

VŠB – Technická univerzita Ostrava  
Fakulta strojní  
Institut dopravy

Optimalizace letové trasy  
Flight Route Optimization

Student:	Pavel Kološ
Vedoucí bakalářské práce:	Ing. Vojtěch Graf

Ostrava 2016

VŠB - Technická univerzita Ostrava  
Fakulta strojní  
Institut dopravy

## Zadání bakalářské práce

Student: **Pavel Kološ**  
Studijní program: B3712 Technologie letecké dopravy  
Studijní obor: 3708R036 Technologie letecké dopravy  
Téma: **Optimalizace letové trasy  
Flight Route Optimization**  
Jazyk vypracování: čeština

Zásady pro vypracování:

1. Úvod
2. Teoretická východiska řešení
3. Analýza vstupních dat pro optimalizační výpočet
4. Návrh řešení
5. Zhodnocení dosažených výsledků
6. Závěr

Seznam doporučené odborné literatury:

1. EISLER, J.: Úvod do ekonomiky dopravy. Praha: CODEX Bohemia s.r.o., 1998. ISBN 80-85963-54-X
2. JANÁČEK, J.: Matematické programování. Žilina: EDIS, 1999. ISBN 80-7100-573-8

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Vojtěch Graf**

Datum zadání: 11.12.2015

Datum odevzdání: 16.05.2016



doc. Ing. Aleš Slíva, Ph.D.  
vedoucí katedry



doc. Ing. Ivo Hlavatý, Ph.D.  
děkan fakulty



### **Místopřísežné prohlášení studenta**

Prohlašuji, že jsem celou bakalářskou práci, včetně příloh, vypracoval samostatně pod vedením vedoucího bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě 7. 4. 2016

A handwritten signature in blue ink, appearing to read 'Karel Paul', with a stylized, cursive script.

Podpis studenta

### ***Poděkování***

*Chtěl bych poděkovat vedoucímu mé bakalářské práce panu Ing. Vojtěchovi Grafovi za jeho trpělivost, vstřícnost a obětavý přístup při řešení problémů spojených s tvorbou této bakalářské práce.*

Prohlašuji, že

- jsem byl seznámen s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- беру на ве́доміі, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě bakalářskou práci užit (§ 35 odst. 3).
- souhlasím s tím, že bakalářská práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užit dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užit své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на ве́доміі, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě 7. 4. 2016



Podpis studenta

Jméno a příjmení autora práce

Pavel Kološ

Adresa trvalého pobytu autora práce

Lamblova 33/56,

Olomouc 779 00

## ANOTACE BAKALÁŘSKÉ PRÁCE

KOLOŠ, P. *Optimalizace letové trasy: bakalářská práce*. Ostrava: VŠB – Technická univerzita Ostrava, Fakulta strojní, Institut dopravy, 2016, 57 s., Vedoucí práce: Graf, V.

Cílem této bakalářské práce je navrhnout matematické modely na bázi lineárního programování, sloužící pro plánování obsluhy letišť z jednoho střediska.

První část práce bude věnována logistickým technologiím, které jsou užívány v letecké dopravě. Dále zde bude uveden proces plánování a přípravy letu.

Další část této práce je zaměřena na teoretické znalosti z oblasti optimalizačních metod a matematického programování potřebné pro tvorbu matematických modelů. Následně budou vytvořeny matematické modely, které budou transformovány do optimalizačního softwaru Xpress-IVE, kde bude ověřena jejich funkčnost.

V závěru budou prezentovány výsledky navržených matematických modelů, včetně zhodnocení dosažených cílů práce.

## ANNOTATION OF BACHELOR THESIS

KOLOŠ, P. *Flight Route Optimization: Bachelor thesis*. Ostrava: VŠB – Technical University Ostrava, Faculty of Mechanical Engineering, Institute of Transport, 2016, 57 s., Thesis head: Graf, V.

The aim of bachelor thesis is to create mathematical models with linear programming using for planning of serving airports from one hub.

The first part of this thesis is dedicated to logistics technologies using in the air transport. There is also mentioned the process of planning and preparing flight.

The next part of this thesis is focused on theoretical knowledge of optimization methods and mathematical programming. The next step will be to create mathematical models which will be transformed to optimization software Xpress-Ive, where it will be tested.

The results of created mathematical models will be presented in the end. The end of thesis including evaluation of achieved aids.

## Seznam zkratk a použitých symbolů

$b_j$	požadavek zákazníka $j$ , pro $j = 1, \dots, n$
$C$	matice vzdáleností
$c_{ij}$	délka minimální letové trasy z letiště $v_i$ na letiště $v_j$ , pro $i = 1, \dots, n$ a $j = 1, \dots, n$ , [km]
$d_j$	dolní mez obsluhy letiště, $v_j$ pro $j = 1, \dots, n$ , [min]
$DEP$	departure (letiště vzletu)
$DEST$	destination (letiště přistání)
$DOHÚF$	dolní odhad hodnoty účelové funkce
$FBW$	Fly By Wire (let vedený elektronickými obvody)
$FMS$	Flight Management System (letový řídicí systém)
$ft$	feet (stopa)
$h_j$	horní mez obsluhy letiště $v_j$ , pro $j = 1, \dots, n$ , [min]
$H$	množina hran
$ICAO$	International Civil Aviation Organisation (Mezinárodní organizace civilního letectví)
$K$	kapacita dopravního prostředku
$k$	stupeň vrcholu (Possova podmínka)
$M$	prohibitivní konstanta
$m$	počet hran grafu
$N$	množina přirozených čísel
$n$	počet letišť (vrcholů)
$NM$	nautical mile (námořní míle)
$o_{ij}$	ohodnocení hran, pro $i = 1, \dots, n$ a $j = 2, \dots, n$
$OPF$	Operation Flight Plan (operační letový plán)
$PPS$	Preflight Planning System (předletový plánovací systém)
$r$	průměrná rychlost letadla, [km·h <sup>-1</sup> ]
$Q$	doba strávená letadlem na letišti, [min]
$SID$	Standard Instrument Departure (odletové tratě)
$SITA$	komunikační síť
$STAR$	Standard Arrival Routes (přiletové tratě)
$st(v_i)$	stupeň vrcholu $v_i$ , pro $i = 1, \dots, n$

$T_{ij}$	doba přesunu letadla z letiště $i$ na letiště $j$ , $pro\ i = 1, \dots, n\ a\ j = 1, \dots, n$ , [min]
$t_j$	čas odletu z letiště $pro\ j = 1, \dots, n$ , [min]
$ÚF$	účelová funkce
$V$	množina vrcholů
$v_i$	letiště (vrchol) $i$ , $pro\ i = 1, \dots, n$
$x_{ij}$	bivalentní proměnná zaručující návaznost obsluhy letišť, $i = 1, \dots, n\ a\ j = 1, \dots, n$
$y_i$	pomocná proměnná zavedená pro potřeby anticyklíci podmínky, $pro\ i = 1, \dots, n\ a\ j = 1, \dots, n$
$\delta_k$	počet vrcholů se stupněm maximálně $k$ , kde $k \in (0, \frac{n}{2})$



# Obsah

Úvod.....	11
1 Úvod do řešené problematiky.....	12
1.1 Logistické technologie a jejich souvislost s plánováním letových tras.....	13
1.2 Předletová příprava .....	14
1.2.1 NAVsystem.....	15
1.2.2 Ostatní software pro plánování a přípravu letu.....	16
2 Optimalizační metody pro plánování okružní letové trasy.....	17
2.1 Exaktní metody .....	17
2.2 Heuristické metody .....	18
2.3 Základní poznatky z teorie grafů určených k nalezení optimálních letových tras	19
2.3.1 Hamiltonova kružnice.....	19
2.4 Plánování okružní letové trasy z jednoho střediska .....	21
2.4.1 Metoda nejbližšího dosud nenavštíveného vrcholu .....	22
2.4.2 Littlův algoritmus .....	22
3 Úvod do matematického programování .....	25
3.1 Lineární programování.....	25
4 Návrh matematických modelů pro řešení vybraných typů úloh.....	28
4.1 Model č. 1 – obsluha letišť bez časových oken.....	29
4.1.1 Formulace problému – model č. 1 .....	29
4.1.2 Rekapitulace použitých zkratk – model č. 1 .....	29
4.1.3 Matematický model č. 1.....	30
4.1.4 Komentář – model č. 1.....	30
4.2 Model č. 2 – obsluha letišť s časovými okny.....	31
4.2.1 Formulace problému – model č. 2 .....	31
4.2.2 Rekapitulace použitých zkratk – model č. 2 .....	32
4.2.3 Matematický model č. 2.....	33
4.2.4 Komentář – model č. 2.....	33

4. 3	Model č. 3 – obsluha letišť s časovými okny a proměnlivou dobou obsluhy .....	34
5	Aplikační část – výpočetní experimenty s navrženými modely .....	35
5. 1	Optimalizační software Xpress – IVE.....	35
5. 2	Optimalizační výpočty v software Xpress - IVE .....	37
5. 3	Výpočetní experiment – model č. 1 .....	39
5. 3. 1	Text programu – model č. 1 .....	40
5. 4	Výpočetní experiment – model č. 2 .....	40
5. 4. 1	Text programu – model č. 2.....	42
5. 5	Výpočetní experiment – model č. 3 .....	43
5. 5. 1	Text programu – model č. 3.....	44
6	Zhodnocení dosažených výsledků .....	46
6. 1	Interpretace výsledků – model č. 1 .....	46
6. 2	Interpretace výsledků – model č. 2 .....	48
6. 3	Interpretace výsledků – model č. 3 .....	50
	Závěr .....	52
	Použitá literatura .....	53
	Seznam obrázků.....	55
	Seznam tabulek .....	57

## Úvod

Rozvoj civilní letecké dopravy lze datovat do období mezi první a druhou světovou válkou. Když byla v roce 1919 společností Aircraft Transport and Travel Limited, předchůdcem dnešních British Airways, zavedena první pravidelná linka mezi Londýnem a Paříží, bylo možné přepravit pouze jednoho cestujícího, avšak hlavním cílem byla přeprava zboží a pošty. Od té doby se letecká doprava posunula ve svém vývoji daleko kupředu. Principy, které fungovaly v první polovině dvacátého století při nízké hustotě provozu a tehdejších technologiích, by dnes nemohly uspět a fungovat.

Letecké společnosti musí pracovat tak, aby byla zajištěna bezpečnost cestujících a celého letu. Jednou z priorit je také finanční zabezpečení leteckého dopravce. Snahou dopravců je ušetřit a minimalizovat náklady na přepravu tak, aby výsledný zisk byl co největší.

Za tím účelem jsou neustále zaváděny nové technologie a systémy, které snižují náklady na přepravu. Příkladem může být Flight Management System (FMS), což je systém, který umožňuje automatizovat širokou škálu letových úkonů a snížit spotřebu paliva. Dalším systémem je systém Fly By Wire (FBW), který slouží k nahrazení manuálního ovládání letadla elektronickým řízením. Díky tomuto systému je letadlo schopno spotřebovávat menší množství paliva. Dalšími přístupy pro snižování spotřeby paliva jsou pojíždění na jednu pohonnou jednotku, start se sníženým výkonem nebo systém WHEELTUG.

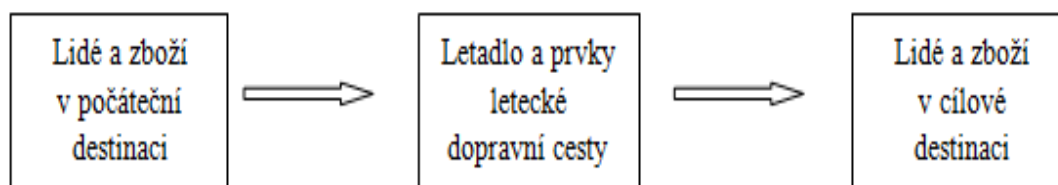
Jednou z oblastí, kde lze minimalizovat náklady, je plánování a volba letové trasy. Plánování optimálních tras mají na starosti operační střediska leteckých dopravců. Při volbě trasy musí být brány v úvahu určité parametry, jako jsou politické důvody, přeletové poplatky nebo bezpečnostní důvody. Jedním z klíčových parametrů je celková délka letové trasy. Převážně společnosti, které se věnují přepravě zboží a zásilek z jednoho střediska na více letišť, musí tento parametr zohlednit, jelikož jen správně naplánovaná trasa může ušetřit finanční prostředky, a tím zvýšit celkový zisk.

Pro plánování letové trasy lze využít optimalizační metody na bázi lineárního programování. Snahou této bakalářské práce je interpretovat a navrhnout matematické modely, které umožní naplánování optimální letové trasy z hlediska vzdálenosti pro obsluhu letišť z jednoho střediska.

# 1 Úvod do řešené problematiky

Letecká doprava patří k dynamicky se rozvíjejícím dopravním odvětvím. Díky své dostupnosti, rychlosti a spolehlivosti se stává stále vyhledávanějším druhem dopravy v přepravě osob a zboží. Vzhledem ke své bezpečnosti je využívána nejen na střední a delší vzdálenosti, ale také na krátké lety. Velkou nevýhodou letecké dopravy je vysoká cena přepravy. Proto je jeden z největších problémů a cílů provozovatelů dosáhnout minimálních nákladů na přepravu tak, aby byla i nadále zajištěna bezpečnost a spolehlivost. [3, 10, 13]

Základním schématem leteckého přepravního systému (obr. 1), je přeprava osob a zboží z letiště vzletu (DEP) do cílové destinace (DEST). K tomu je využit dopravní prostředek – letadlo a dopravní cesta neboli systém letových tras. [3]

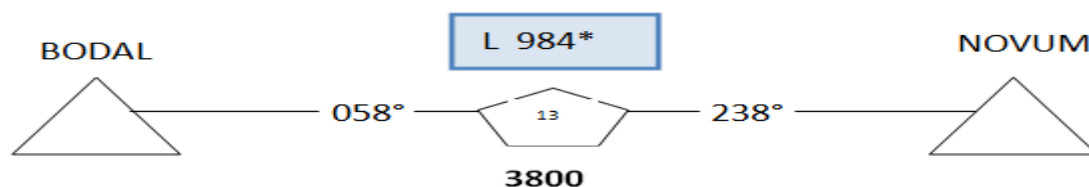


Obr. 1 – Základní schéma leteckého přepravního systému [3]

Letové tratě, po kterých se let uskutečňuje, lze rozdělit do tří kategorií:

- **SID tratě** – jsou odletové tratě z letiště vzletu, které se používají pro efektivní odlety letadel z letišť a tyto tratě končí vždy na prvním bodu TRIP tratě,
- **TRIP tratě** – jsou tratě, které jsou tvořeny množinou bodů a letových cest, po kterých letadlo letí na konečný bod TRIP tratě,
- **STAR tratě** – jsou příletové tratě na cílové letiště, které navazují na poslední bod TRIP tratě. [11, 12, 13]

Systém letových tras je tvořen traťovými body a letovými cestami. Příklad je uveden na obr. 2. Na letové trati, s označením L984\* jsou dva traťové body – BODAL a NOVUM. Jestliže je let uskutečněn z bodu BODAL na bod NOVUM, je magnetický kurz 058° a při letu z bodu NOVUM na bod BODAL 238°. Dalším údajem je délka trati, která je v tomto případě 13 NM. Pod údajem o délce trati je minimální povolená výška letu, která je v daném úseku 3800 ft. V některých případech je na trati uveden i směr letu, jestliže se jedná o jednosměrné trati.



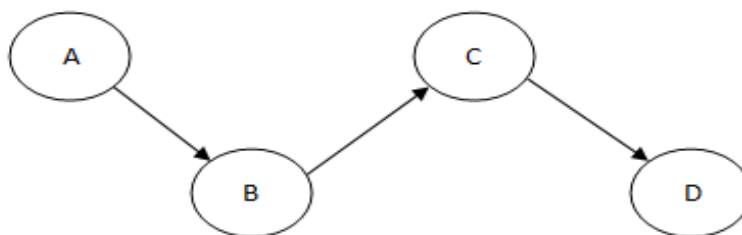
Obr. 2 – Úsek letové trati L984\*

## 1.1 Logistické technologie a jejich souvislost s plánováním letových tras

Logistika je definována jako vědní disciplína obsahující metody, díky kterým lze optimalizovat dopravní proces na základě logistických metod a technologií za podpory softwarového zpracování informací. [3]

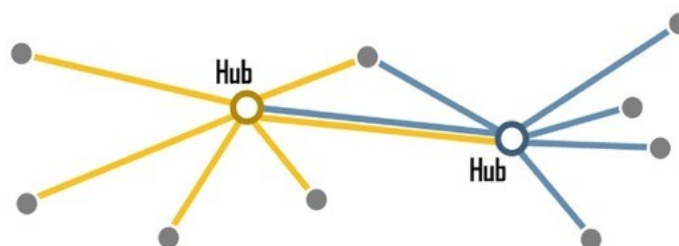
V letecké dopravě se využívají dvě základní logistické technologie:

- **point to point** – neboli liniová přeprava, kdy lze pomocí jedné linky obsloužit několik letišť – obr. 3,



Obr. 3 – Princip technologie point to point [13]

- **hub and spoke** – kde hub představuje uzel, který je používán, jako přestupní místo, ze kterého pokračují cestující nebo zboží návaznými spoji (leteckou, silniční nebo železniční dopravou) do cílových destinací – obr. 4.



Obr. 4 – Princip technologie hub and spoke [18]

Nedílnou součástí plánování letu, ať již v rámci technologie point to point nebo hub and spoke, je předletová příprava.

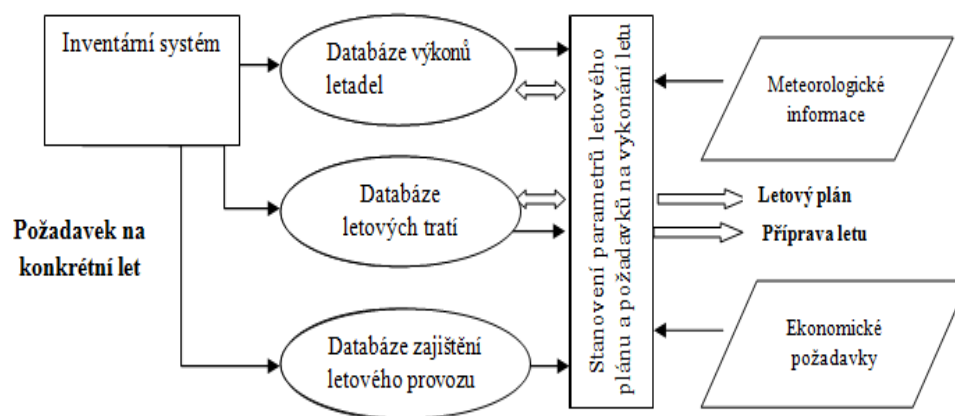
## 1.2 Předletová příprava

Na předletovou přípravu jsou kladeny vysoké nároky a požadavky. Hlavními úkoly plánování a přípravy letu jsou:

- stanovení cílového letiště, které splňuje dané požadavky (vhodnost dráhového systému, dostupnost služeb, radionavigační vybavení, provozní doba, aj.)
- výběr vhodné trati letu vzhledem k ekonomickým aspektům,
- určení letové posádky a její komplexní příprava na plánovaný let,
- zpracování a podání ICAO letového plánu,
- stanovení vybraných parametrů letadla (rychlosti, výšky, palivo, aj.) pro spolehlivost a bezpečnost letu.

Výstupem procesu plánování a přípravy letu jsou parametry (hmotnost, spotřeba, vyvážení, rychlost) pro vybrané letadlo, které jsou zpracovány do tzv. operačního letového plánu (OFP), který se uschovává po dobu tří let. Kvalitně a komplexně vyškolená posádka je pro spolehlivý a bezpečný let nutností. Pomocí výpočtů, při nichž se vychází z navigačních parametrů, výkonových charakteristik daného typu letadla a aktuálního meteorologického stavu, dochází k hledání a zvolení optimální letové cesty a profilu letu. V průběhu procesu optimalizace jsou získávány informace z databáze letových tratí, databáze navigačních zařízení a databáze hlasných bodů a letišť. [3, 11, 13]

Struktura systému plánování a přípravy letu je na obr. 5. Na daném schématu jsou zahrnuty všechny požadavky a zadané parametry. Z inventárního systému jsou vybrána data z databáze výkonů letadel, letových tratí a databáze zajištění letového provozu. Po zvolení těchto aspektů dochází ke stanovení parametrů letového plánu a požadavků na vykonání letu. Tyto parametry jsou ovlivněny meteorologickými informacemi a dále ekonomickými požadavky. Po kompletním zpracování všech potřebných podkladů, je možné vytvořit operační a ICAO letový plán a finální přípravu letu.



Obr. 5 – Schéma systému plánování a přípravy letu [3]

Předpokladem kvalitního naplánovaného letu je kvalifikovaný personál operačního oddělení, který s danými parametry pracuje a zpracovává pro posádku podklady a informace o plánovaném letu. Letečtí dopravci využívají softwary, které mají za úkol minimalizovat náklady na přepravu a naplánovat spolehlivý a bezpečný let – dále v podkapitolách 1. 2. 1 a 1. 2. 2.

### 1. 2. 1 NAVsystem

Mezi užívané navigační systémy pro plánování letu patří NAVsystem od společnosti NAV Flight Services. Je napojen na komunikační síť a na internet. Automaticky zpracovává příchozí zprávy a odpovídá na ně. Po síti SITA umožňuje komunikaci s posádkou letadla za letu. Letecké společnosti poskytuje statisticky zpracovanou zpětnou vazbu (využití posádek, letadel, spotřeba paliva, atd.) pro optimalizaci nákladů pro každý let, což má dopad na zlepšení hospodaření společnosti. [3]

NAVsystem je určený obchodním a soukromým provozovatelům, kteří mají vlastní operační oddělení, a také pro společnosti nabízející komplexní plánování letu. [3]

NAVsystem je tvořen z několika složek:

- **NAVairways** – databáze traťových bodů, letových cest a letišť. Provozovatel má možnost využít komerční navigační data nebo si potřebné parametry zvolí sám. Součástí NAVairways je elektronická mapa *NAVstar*, kde jsou veškerá omezení a informace o letových tratích, což umožňuje efektivní volbu tratě. Systém umožňuje jednoduché plánování letů a vyhledávání optimálních letových cest a profilů plánovaného letu. Na obr. 6 je možno vidět pracovní prostředí systému NAVairways

s uvedením letových cest, traťových bodů, map letových oblastí SID a STAR tratí.



Obr. 6 – Ukázka pracovního prostředí systému NAVairways [17]

- **NAVplan** – představuje databázi výkonů letadel, což zahrnuje hodnoty spotřeby paliva a rychlosti letu v závislosti na meteorologických situacích v jednotlivých fázích letu,
- **NAVlog** – provádí optimalizační výpočty, jejichž výsledkem jsou podklady pro předletovou přípravu a provozní letový plán (OFP), který je základním dokumentem předletové přípravy a zároveň základní navigační příprava posádky. Systém automaticky generuje ICAO letový plán, který je odeslán ke schválení do Central Flow Managment Unit. [3]

### 1. 2. 2 Ostatní software pro plánování a přípravu letu

Kromě NAVsystemu jsou používány i další navigační software. Patří zde například:

- **Preflight Planning System PPS** – od společnosti Air Support. Tento systém umí po zadání veškerých údajů vytvořit optimalizovaný OFP, který je automaticky odeslán do systému řízení a optimalizace letu FMS,
  - **Jeppesen Flight Planning (JetPlan)** – od firmy Jeppesen. Patří mezi nejpoužívanější pro profesionální plánování letů. Systém je napojen na celou řadu aplikací, včetně celosvětové struktury letových cest,
  - **Amadeus Altéa Customer Managment Solution** – pro plánování a kompletní přípravu letu,
  - **RM Rocode** – od firmy Sabre Airline Solutions, který je užíván malými nízkonákladovými společnostmi, provozující lety na střední vzdálenosti.
- [3]



## 2 Optimalizační metody pro plánování okružní letové trasy

Úkolem optimalizačních metod je zajistit co nejvýhodnější řešení daného problému. Na optimalizační metody jsou kladeny dva základní požadavky, a to:

- **spolehlivost výpočtu** – schopnost nalézt optimální řešení,
- **efektivita výpočtu** – rychlost nalezení optimálního řešení.

Na začátku procesu řešení s využitím optimalizačních metod je stanoveno optimalizační kritérium – hledisko posuzování efektivity řešení, které bývá nejčastěji vyjádřeno pomocí účelové funkce (ÚF). Účelová funkce je vyjádřena funkčním předpisem a umožňuje srovnávat kvalitu jednotlivých řešení z pohledu optimalizovaného kritéria. Při optimalizačních úlohách je řešitel zpravidla omezen určitými faktory, což jsou vstupní limity pro řešení. Jestliže u nalezeného řešení není některý ze stanovených faktorů dodržen, je takové řešení považováno za nepřipustné. Každé řešení, které vyhovuje všem omezujícím podmínkám, nazýváme přípustné řešení. Cílem procesu optimalizace je dosáhnout optimálního řešení, které minimalizuje nebo maximalizuje hodnotu účelové funkce za daných omezujících podmínek. V praxi se vyskytují úlohy, které mají více než jedno optimální řešení, avšak na druhé straně jsou případy, kdy úloha optimálního řešení nemá. Pro zjištění, zda je dané řešení optimální či nikoliv, se používá test optimality. Každá optimalizační metoda má svůj test optimality a v reálu neexistuje žádný univerzální test optimality, který by bylo možné aplikovat na všechny úlohy. Tak jako neexistuje univerzální test optimality, neexistuje ani univerzální optimalizační metoda. Metoda, která v jednom případě zaručí optimální řešení, nemusí najít optimální řešení i v jiné úloze. [4, 6]

Optimalizační metody lze rozdělit do dvou kategorií – na metody exaktní a metody heuristické – dále podkapitoly 2. 1 a 2. 2.

### 2. 1 Exaktní metody

Exaktní metody optimalizace, jsou podloženy poznáním exaktních vědních oborů (nejčastěji matematických disciplín). Jejich největší výhodou je spolehlivost. Exaktní metody zaručují nalezení optimálního řešení v případě, že takové řešení existuje (tzn., že řešením není prázdná množina). Mají však jednu nevýhodu – ne vždy jsou dostatečně výkonné, aby našly optimum v požadovaném čase. [4, 6, 9]

Pro tyto metody obecně je zpravidla stanovený tzv. iterační postup. Jedná se o formální postup, kdy budou algoritmy vždy od jednoho řešení přecházet ke druhému. [6]

Exaktní metody lze rozdělit:

- **na metody matematické statistiky** – používané pro uskutečnění analýzy posuzovaných procesů a pro zjištění kvantitativních charakteristik,
- **metody operačního výzkumu (operační analýzy)** – souhrn metod, které pomocí matematických disciplín modelují určité stavy technologických nebo rozhodovacích procesů. Patří zde:
  - *matematické modely* – simulační, analytické, grafické,
  - *teorie grafů* – pro stanovení optimální cesty, optimální struktury, optimálních toků v grafech,
  - *teorie hromadné obsluhy* – pro řešení úloh technologie práce v dopravě a přepravě, nebo skladech,
  - *synergetika* – pro vzájemné vazby procesů. [4]

## 2.2 Heuristické metody

Tyto metody jsou využívány v rozhodovacích procesech s vysokou mírou neurčitosti a v úlohách, kde není možné využít exaktní metody k nalezení řešení ve stanoveném čase. Výhodou heuristických metod je rychlost řešení, jelikož jsou na řešení obecně jednodušší než exaktní metody. Jejich největší nevýhodou je, že nezaručují nalezení optimálního řešení. [4, 6, 8]

Heuristické metody lze také rozdělit do dvou skupin:

- **na expertní systémy** – programové produkty, které jsou založené na znalostech, vedoucích k vyřešení úlohy,
- **na metody tvořivého myšlení** – systémy, které využívají tvůrčí schopnosti expertů, a jejichž základním principem je výběr, přetváření a spojování prvků předcházejících zkušeností. [4]

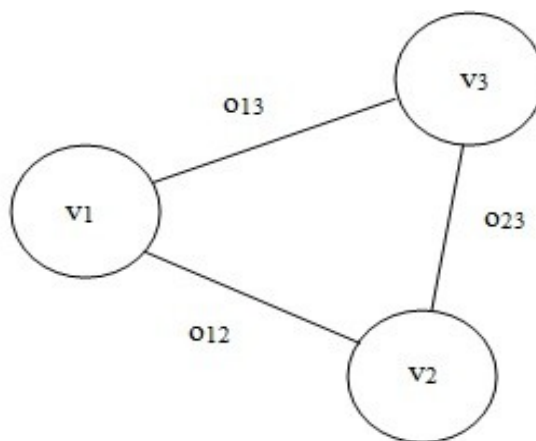
Než bude věnována pozornost jednotlivým optimalizačním metodám pro plánování okružní letové trasy, což je předmětem práce, je potřeba shrnout základní pojmy z teorie

grafů – kapitola 2. 3, jejíž znalost je pro pochopení a práci s jednotlivými metodami a algoritmy nezbytná.

## 2. 3 Základní poznatky z teorie grafů určených k nalezení optimálních letových tras

Pro lepší vizuální znázornění vazeb v dopravních systémech se často využívá diagramů grafů. Grafy jsou tvořeny množinou vrcholů  $V$ , kde  $n$  je počet vrcholů grafu a množinou hran  $H$ , kde  $m$  je počet hran grafu. Vrcholům nebo hranám mohou být přiřazena ohodnocení, přičemž ohodnocení vrcholů nejčastěji reprezentuje počet obsluh daného segmentu dopravní sítě a ohodnocení hran nejčastěji představuje jejich délky.

Graf znázorněný na obr. 7 je tvořen vrcholy  $V=\{v_1, v_2, v_3\}$ , hranami  $H=\{h_{12}, h_{23}, h_{13}\}$ . Ohodnocení hran představují délky hran -  $o_{12}, o_{23}, o_{13}$ . [2, 5]



Obr. 7 – Graf s ohodnocením hran

V praxi lze využít řešení pomocí grafů v různých situacích. V případě řešené práce se jedná o řešení letecké distribuční sítě, kde vrcholy představují letiště a hrany jsou letové trasy mezi jednotlivými vrcholy, přičemž ohodnocení hran reprezentují vzdálenosti mezi dvěma letišti.

V předložené bakalářské práci je řešena problematika optimalizace letové trasy, která je založena na vyhledávání minimální Hamiltonovy kružnice v zadané dopravní síti.

### 2. 3. 1 Hamiltonova kružnice

V teorii grafů se kružnicí rozumí pravidelný graf, v němž s každým vrcholem sítě incidují právě dvě hrany - vrcholy grafu jsou druhého stupně. Hamiltonova kružnice, je faktorový podgraf (obsahuje všechny vrcholy původního grafu) druhého stupně.

Obecně neplatí, že by každý graf obsahoval Hamiltonovu kružnici. V minulosti proto byly pro ověření existence Hamiltonovy kružnice formulovány určité podmínky. Ty lze rozdělit na nutné a postačující. Nutné podmínky existence Hamiltonovy kružnice v neorientovaném grafu:

- graf musí být **souvislý** (mezi každou dvojicí vrcholů je cesta, která je spojuje) a **konečný**,
- musí obsahovat minimálně **tři vrcholy**,
- nesmí obsahovat **mosty** (hrana, jejímž odstraněním se zvýší počet komponentů grafu o jeden) a **artikulace** (vrchol, jehož odstraněním se zvýší počet komponentů grafu alespoň o jeden). [5]

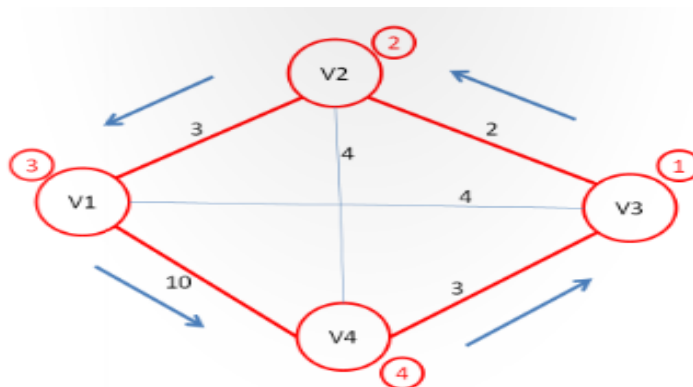
Jestliže není některá z nutných podmínek splněna, víme, že Hamiltonova kružnice v dané síti neexistuje. Avšak ani splnění nutných podmínek nezaručí existenci Hamiltonovy kružnice v síti. [4]

Při ověřování existence Hamiltonovy kružnice slouží ještě tzv. postačující podmínky:

- **Diracova podmínka** – Hamiltonova kružnice v grafu existuje, když má každý vrchol stupeň alespoň  $\frac{n}{2}$ , kde  $n$  je počet vrcholů grafu. Matematicky vyjádřeno:  $st(v_i) \geq \frac{n}{2}$ , pro  $i=1, \dots, n$ ,
- **Oreho podmínka** – Hamiltonova kružnice v grafu existuje, když pro každou dvojici nesousedících vrcholů  $v_i$  a  $v_j$  platí, že součet jejich stupňů je alespoň  $n$ . Matematicky vyjádřeno:  $st(v_i) + st(v_j) \geq n$
- **Posova podmínka** – Hamiltonova kružnice v grafu existuje, když je splněno, že  $\delta_k < k$ , kde  $k \in N$  ( $N$  je množina přirozených čísel) a současně  $k \in (0, \frac{n}{2})$ ,  $\delta_k$  je počet vrcholů se stupněm maximálně  $k$  a  $n$  je počet vrcholů sítě. [5]

Postačující podmínky jsou ve vzájemném vztahu. Splnění Diracovy podmínky, znamená také splnění Oreho a Posovy podmínky. Splnění Oreho podmínky znamená automaticky splnění Posovy podmínky, avšak neznamená splnění Diracovy podmínky. Při splnění Posovy podmínky, není automaticky splněna Diracova a Oreho podmínka. Nesplnění žádné postačující podmínky neznamená neexistenci Hamiltonovy kružnice

v síti, kde jsou všechny nutné podmínky splněny. Na obr. 8 je vidět příklad Hamiltonovy kružnice, kde počáteční i koncový vrchol je  $v_3$ . [6]



Obr. 8 – Hamiltonova kružnice [7]

## 2. 4 Plánování okružní letové trasy z jednoho střediska

Plánování obsluhy vrcholů dopravní sítě z jednoho střediska je v odborné literatuře často nazývána tzv. úlohou obchodního cestujícího. Cílem je najít minimální Hamiltonovu kružnici, což je Hamiltonova kružnice, u které je součet ohodnocení hran minimální. Obsluha probíhá tak, že dopravní prostředek zahájí obsluhu sítě ve středisku, v rámci jedné okružní jízdy obslouží všechny zákazníky a poté se vrací zpět do střediska. [16]

Obsluha je realizována v atrakčním obvodu, což je část sítě obsluhované z jednoho střediska. Optimalizačním kritériem je absolvovaná vzdálenost, kterou obslužné vozidlo urazí. Limitujícím faktorem může být kapacita vozidla, která není neomezená. Chceme-li v rámci jedné okružní jízdy obsloužit všechny zákazníky, musí platit vztah:

$$\sum_{j=1}^n b_j \leq K ,$$

kde  $K$  je kapacita dopravního prostředku a  $b_j$  pro  $j= 1, \dots, n$  jsou požadavky jednotlivých zákazníků. [5]

Při praktických výpočtech se minimální Hamiltonova kružnice hledá v transformované síti. Transformací získáme graf, který je tvořen stejnou množinou vrcholů, jako graf původní. Hrany představují minimální cesty mezi jednotlivými vrcholy. Transformace se provádí, protože transformovaný graf je úplný (kompletní) a v takovém grafu existuje Hamiltonova kružnice vždy. Navíc v dopravní praxi může být za účelem úspory ujeté vzdálenosti naplánována taková trasa, která ve skutečnosti některými vrcholy prochází

vícekrát. Po vyhledání minimální Hamiltonovy kružnice, je nutno tuto kružnici transformovat zpět do původní sítě. [4]

Pro řešení minimální Hamiltonovy kružnice (úlohy obchodního cestujícího) lze použít různé exaktní nebo heuristické metody a algoritmy – metoda nejbližšího dosud nenavštíveného vrcholu a Littlův algoritmus – viz následující kapitoly 2. 4. 1 a 2. 4. 2.

#### **2. 4. 1 Metoda nejbližšího dosud nenavštíveného vrcholu**

Jedná se heuristickou metodu sloužící k nalezení Hamiltonovy kružnice v grafu. Vzhledem k tomu, že se jedná o heuristickou metodu, není ve výsledku zaručeno nalezení optimálního řešení, avšak to neznamená, že optimální řešení nemůže být použitím metody nejbližšího dosud nenavštíveného vrcholu nalezeno. Velkou výhodou této metody je její rychlost a jednoduchost, takže má smysl tuto metodu aplikovat tehdy, není-li z časových důvodů možné využít některou z exaktních metod. [5, 6]

Algoritmus řešení:

1. Je vybrán libovolný vrchol a konstrukce Hamiltonovy kružnice je započata tak, že zařadíme hranu incidentující s tímto vrcholem s nejmenším ohodnocením. Je-li takových hran více, je vybrána libovolná z nich. Pokračuje se krokem 2.
2. Je-li vybráno  $n-1$  hran, následuje krok 4. V opačném případě se pokračuje krokem 3.
3. Z naposledy navštíveného vrcholu, je jako další zařazena do posloupnosti hran hrana incidentující s dosud nenavštíveným vrcholem, která má minimální ohodnocení hran. Existuje-li více takových hran, je vybrána libovolná z nich a následuje návrat na krok 3.
4. Po obsloužení všech naplánovaných vrcholů se vracíme do výchozího vrcholu a algoritmus končí. Nalezená kružnice je Hamiltonova.

#### **2. 4. 2 Littlův algoritmus**

Mezi exaktní metody sloužící k nalezení minimální Hamiltonovy kružnice patří Littlův algoritmus, který zaručuje nalezení optimálního řešení v dopravní síti. Jeho nevýhodou je značná náročnost při řešení. Algoritmus je založen na metodě větví a mezí – množina řešení se dělí na menší podmnožiny. Všechna řešení v dané podmnožině jsou dána dolním odhadem účelové funkce. Ke znázornění výpočetního postupu se používá graf – strom řešení. Vstupem do Littlova algoritmu je matice vzdáleností mezi jednotlivými vrcholy

sítě. Po skončení výpočtu je zapotřebí vzdálenosti zahrnuté do optimálního řešení transformovat zpět do původní dopravní sítě. [4, 7]

Algoritmus řešení:

1. V každém řádku matice vzdáleností se odečte řádkové minimum tak, aby v každém řádku byla získána alespoň jedna nula. Pokračuje se krokem č. 2.
2. V řešící matici provedeme stejnou operaci u každého sloupce a následuje postup na krok č. 3.
3. Hodnoty, které byly odečteny v krocích č. 1 a 2, udávají po sečtení dolní odhad hodnoty účelové funkce (DOHÚF). Následuje krok č. 4.
4. V řešící tabulce je přiřazeno každé nule ohodnocení, které je rovno součtu řádkového a sloupcového minima. Nelze-li nulu ohodnotit, pokračuje se krokem č. 10, jinak krokem č. 5. Ohodnocení nuly představuje nárůst DOHÚF.
5. Je vybrána nula s maximálním ohodnocením, což znamená, že navrhujeme zařadit příslušnou hranu do Hamiltonovy kružnice. Současně je aktualizován a upravován strom řešení. Pokračuje se krokem č. 6.
6. Je zrušeno ohodnocení nul a následuje krok č. 7
7. Dochází k redukci řešící tabulky o řádek a sloupec, ve kterém leží nula, která byla vybrána, následuje krok č. 8.
8. V řešící tabulce jsou zakázány prvky, kde by výběrem těchto prvků mohlo vzniknout nepřípustné řešení (předčasné uzavření Hamiltonovy kružnice). Následuje postup na krok č. 9.
9. Provedeme kontrolu, zda je v každém řádku a sloupci alespoň jedna nula. Pokud ne, odečteme řádkové či sloupcové minimum, čímž dojde k navýšení DOHÚF. Vracíme se ke kroku č. 4.
10. Pokud v řešící tabulce nelze nuly ohodnotit, pak tyto nulové prvky uzavírají Hamiltonovu kružnici. Pokud se ve stromu řešení nachází větev s hodnotou DOHÚF nižší, než je hodnota DOHÚF posledního řešení, musí se tato větev prohledat a následuje návrat na krok č. 1. V opačném případě se pokračuje na krok č. 11.
11. Poslední nalezené řešení je optimální. Je nalezena minimální Hamiltonova kružnice.

Littlův algoritmus je nejvhodnější metodou pro řešení obsluhy letišť z jednoho střediska, protože ve výsledku zaručuje optimální řešení. Vzhledem k tomu, že v mé práci se jedná o obsluhu pěti vrcholů, nepředpokládá se ani časová náročnost na řešení.

V následující kapitole bude stručně charakterizováno matematické programování, díky kterému lze modelovat různé dopravní situace pomocí matematických modelů, včetně úlohy o nalezení minimální Hamiltonovy kružnice – tzn. nalezení optimální letové trasy.



### 3 Úvod do matematického programování

Matematické programování patří ve srovnání s jinými matematickými obory mezi poměrně mladý obor. Jeho vznik se datuje přibližně do padesátých let minulého století a je úzce spojen s rozvojem výpočetní techniky. Postupem času proniklo matematické programování do různých odvětví vědy, techniky, ekonomiky, nebo dopravy. V současnosti se lze s využitím matematického programování setkat při organizaci všech druhů dopravy, při plánování výroby, při přesunu a zásobování vojenských jednotek a v mnoha dalších činnostech. [1]

Při matematickém programování je třeba provést rozhodnutí při řízení procesu, nebo systému, jehož výsledkem by mělo být nové řešení, které zefektivní danou činnost. Takovým rozhodnutím může být například plánování přepravy určitého zboží ze skladu k zákazníkům tak, aby byly celkové náklady na přepravu minimální.

Předmětem matematického programování je tvorba matematických modelů, které představují reálné situace. Matematické modelování patří mezi exaktní metody a lze ho dělit podle mnoha kritérií:

- **podle náhodné proměnné** – rozlišujeme *deterministické modely* (bez náhodných proměnných) a *stochastické modely* (s náhodnými proměnnými),
- **podle časové proměnné** – rozlišujeme *statické modely* (řešení není proměnné v čase, jedná se o rozhodnutí v konkrétní situaci) a *dynamické modely* (řešení je proměnné v čase, umožňují učinit výhledové řešení),
- **podle vlastností funkcí** – dělíme modely na *lineární* (podmínky úlohy jsou vyjádřeny lineárními rovnicemi a nerovnicemi a účelová funkce je lineární) a *nelineární* (podmínky jsou tvořeny nelineárními rovnicemi či nerovnicemi nebo se nelinearita vyskytuje v účelové funkci). [4, 8]

#### 3.1 Lineární programování

Lineární programování je vhodným prostředkem pro procesní rozhodování a je nejlépe zpracovanou částí matematického programování. Jeho výhodou je, že při dostatku času a výpočetního výkonu nám zpravidla zaručuje nalezení optimálního řešení.

Lineární matematické modely jsou tvořeny dvěma druhy veličin:

- **konstantami** – vstupní veličiny, které jsou vloženy do modelu, a jejich hodnota se nemění,
- **proměnnými** – vstupní veličiny, kterými modelujeme rozhodnutí, a jejichž hodnota se během výpočtu mění. [8]

Počet konstant v modelu je neomezený a závisí pouze na řešiteli, kolik jich do modelu zavede. Naopak počet proměnných v modelu je vázán na počet rozhodnutí, která mají být uskutečněna. Do některých matematických modelů je třeba kromě klasických rozhodovacích proměnných zavést pomocné proměnné.

Každá proměnná musí mít přiřazený definiční obor. V lineárním programování rozlišujeme tři druhy definičních oborů:

- **nezáporných reálných čísel**,
- **nezáporných celých čísel**,
- **bivalentních hodnot** – množina hodnot 0 nebo 1. [4]

Každý lineární matematický model se skládá ze dvou částí:

- **z účelové funkce**,
- **soustavy omezujících podmínek**.

Účelová funkce vyjadřuje optimalizační kritérium, s jehož pomocí se vypočítá výsledná hodnota optimalizované veličiny. Cílem je získat optimální řešení, které je závislé na typu zvoleného extrému – *minimum* (například minimalizace nákladů na přepravu) a *maximum* (příkladem může být zvýšení objemu přepravy nebo zisk).

Soustava omezujících podmínek, vyjadřující množinu přípustných řešení, dělíme na dvě skupiny:

- **obligatorní podmínky**,
- **strukturální podmínky**. [4]

Obligatorní podmínky reprezentují definiční obory proměnných vyskytujících se v modelu. Počet obligatorních podmínek je roven počtu proměnných vystupujících v modelu.

Strukturální podmínky v modelu reprezentují reálná omezení nebo vytvářejí vazby mezi jednotlivými proměnnými, ať už rozhodovacími nebo pomocnými. Počet strukturálních podmínek je závislý na množství reálných omezení a počtu vazeb.

Řešením úloh lineárního programování mohou být tyto tři situace:

- **úloha má optimální řešení** – úloha může mít jedno nebo více optimálních řešení,
- **úloha nemá optimální řešení** – množina přípustných řešení je prázdná, protože v úloze nelze splnit všechny podmínky,
- **optimální řešení nelze najít** – množina přípustných řešení je neohraničená ve směru optimalizace.

Při vytváření matematických modelů lineárního programování musí být dodržována určitá pravidla. Jejich nedodržení způsobí vznik nelineárních modelů, jejichž řešení je obtížnější nebo není možné, protože není k dispozici vhodná metoda. Obecně při sestavování matematických modelů platí dvě pravidla:

- proměnné je mezi sebou možné sčítat, odčítat, nebo násobit reálnou konstantou (nelze proměnné násobit mezi sebou),
- při tvorbě omezujících podmínek je povoleno používat relační znaménka  $=, \geq, \leq$ . [15]

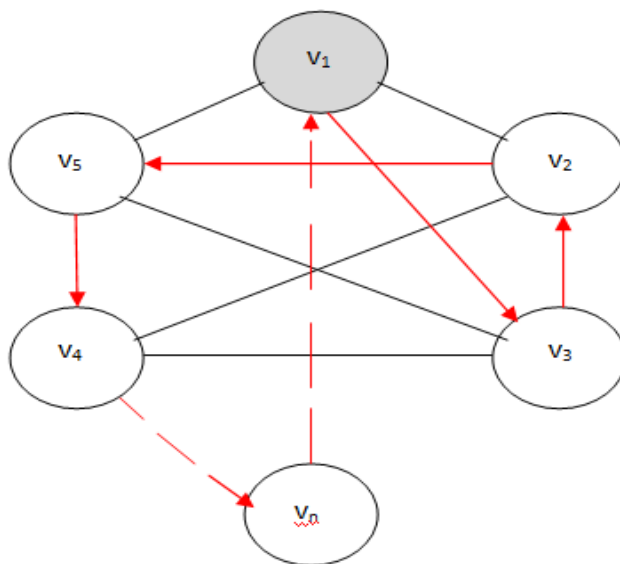
V lineárním programování není zvláštností, že na jeden typ úlohy lze použít více modelů a typů řešení. V případech, kdy je možnost řešit úlohu více způsoby, vybírá řešitel takový model, který je pro řešení zadané úlohy nejefektivnější.

Následující kapitola obsahuje návrh matematických modelů pro optimalizaci okružní letové trasy.

## 4 Návrh matematických modelů pro řešení vybraných typů úloh

V této kapitole budou prezentovány tři matematické modely pro naplánování obsluhy letišť z jednoho střediska. Cílem všech uvedených modelů je optimalizace letové trasy. Budou uvažovány tři případy – tzv. úloha „bez časových oken“, úloha s „časovými okny“ a úloha s „časovými okny“ a proměnlivou dobou obsluhy. V prvním modelu (kap. 4. 1), je úkolem minimalizace délky absolvované trasy bez ohledu na čas obsluhy letišť. Druhý model (kap. 4. 2), kromě minimalizace celkové absolvované vzdálenosti, zohledňuje časové intervaly, ve kterých musí být realizována obsluha zákazníka na letištích. Ve třetím modelu (kap. 4. 3.), bude upravena časová konstanta  $Q$ , která se bude měnit v závislosti na době obsluhy letiště. Ve všech třech modelech budou uvažovány situace, při kterých je kapacita letadla dostatečná na to, aby splnila požadavky jednotlivých letišť, takže ji lze pro další výpočty zanedbat.

Na obr. 9 je znázorněna okružní letová trasa vztahující se ke všem třem modelům, kde probíhá obsluha množiny letišť ( $|I| = n$ ), z výchozího letiště  $v_1$ , přičemž po obsluze všech letišť v rámci jednoho okružního letu následuje návrat na letiště  $v_1$ .



Obr. 9 – Obsluha letišť ze střediska  $v_1$

V aplikační části budou všechny navržené modely transformovány do jazyka Mosel a následně bude v optimalizačním software Xpress-IVE ověřena jejich funkčnost a budou vyhodnoceny dosažené výsledky.

## 4. 1 Model č. 1 – obsluha letišť bez časových oken

První model je koncipován tak, aby jeho výstupem byla optimalizovaná letová trasa, kde budou v rámci jednoho okružního letu obsloužena všechna letiště, a jejíž celková délka bude minimální. V tomto modelu se neuvažuje s časovými okny pro obsluhu jednotlivých letišť.

### 4. 1. 1 Formulace problému – model č. 1

Je dána síť letových tras obsahující množinu letišť  $V$  ( $|V| = n$ ), ve které má probíhat zasilatelská činnost (svoz a rozvoz zásilek) leteckou dopravou. Letiště  $v_1$  představuje distribuční středisko, ze kterého bude obsluha probíhat. Dále je známa matice vzdáleností  $C$ , přičemž prvek  $c_{ij}$  reprezentuje délku minimální letové trasy z letiště  $v_i$  pro  $i = 1, \dots, n$  na letiště  $v_j$  pro  $j = 1, \dots, n$ . Úkolem je naplánovat trasu letu tak, aby v rámci jednoho okružního letu byla navštívena všechna letiště  $v_2, v_3, \dots, v_n$  a zároveň, aby výchozí a koncový bod trasy byl ve vrcholu  $v_1$ .

Optimalizačním kritériem, je celková vzdálenost, kterou letadlo absolvuje při obsluze naplánované množiny spojů. Cílem optimalizačního výpočtu je minimalizace celkové vzdálenosti. Návaznost mezi jednotlivými letišti je modelována bivalentní proměnnou  $x_{ij}$ . Je-li  $x_{ij} = 0$ , není spoj realizován, ale při  $x_{ij} = 1$  je spoj z letiště  $i$  na letiště  $j$  realizován. Dále je do modelu zavedena pomocná proměnná  $y_i$  pro  $i = 2, \dots, n$ , která se využívá v tzv. anticyklíci podmínce, tedy podmínce zajišťující, že nedojde k rozpadu okružního letu na více částí.

### 4. 1. 2 Rekapitulace použitých zkratk – model č. 1

Konstanty:

- $c_{ij}$  délka minimální letové trasy z letiště  $v_i$  na letiště  $v_j$ ,  
*pro  $i = 1, \dots, n$  a  $j = 1, \dots, n$ ,*
- $n$  počet letišť.

Proměnné:

- $x_{ij}$  proměnná modelující návaznost obsluhy jednotlivých letišť,  
*pro  $i = 1, \dots, n$  a  $j = 1, \dots, n$ ,*
- $y_i$  pomocná proměnná zavedená pro potřeby anticyklíci podmínky,  
*pro  $i = 1, \dots, n$ .*

#### 4. 1. 3 Matematický model č. 1

$$\min f(x, y) = \sum_{i=1}^n \sum_{j=1}^n c_{ij} \cdot x_{ij} \quad (4.1.1)$$

$$\sum_{i=1}^n x_{ij} = 1, \text{ pro } j = 1, \dots, n \quad (4.1.2)$$

$$\sum_{i=1, i \neq j}^n x_{ij} = \sum_{i=1, i \neq j}^n x_{ji}, \text{ pro } j = 1, \dots, n \quad (4.1.3)$$

$$y_i - y_j + n \cdot x_{ij} \leq n - 1, \text{ pro } i = 2, \dots, n \text{ a } j = 2, \dots, n \quad (4.1.4)$$

$$x_{ij} \in \{0, 1\}, \text{ pro } i = 1, \dots, n \text{ a } j = 1, \dots, n \text{ a } i \neq j \quad (4.1.5)$$

$$y_i \geq 0, \text{ pro } i = 2, \dots, n \quad (4.1.6)$$

#### 4. 1. 4 Komentář – model č. 1

Rovnice (4.1.1) je účelovou funkcí reprezentující celkovou délku absolvovaných letových tras. Když proměnná  $x_{ij} = 1$ , potom se hodnota  $c_{ij}$  do účelové funkce započítá, v opačném případě se nezapočítá.

Skupina omezujících podmínek (4.1.2) zajistí, že každý spoj bude obslužen právě jednou v rámci jednoho okružního letu. Skupina omezujících podmínek (4.1.3) zajistí kontinuitu organizace letu v rámci každého letiště. Skupina omezujících podmínek (4.1.4) zajistí, že nedojde k rozpadu vytvářeného cyklu na více částí. Důkaz nerovnice (4.1.4) je proveden na čtyřech vrcholech. Pro řešení jsou vybrány pouze ty nerovnice, kde je spoj realizován, tedy ty, kde bivalentní proměnná  $x_{ij} = 1$ .

Důkaz nerovnice (4.1.4):

Při důkazu nejprve sestavíme nerovnice pro případy, kdy  $x_{ij} = 1$ :

$$y_2 - y_3 + n \cdot x_{23} \leq n - 1,$$

$$y_3 - y_4 + n \cdot x_{34} \leq n - 1,$$

$$y_4 - y_2 + n \cdot x_{42} \leq n - 1,$$

a vzhledem k tomu, že počet vrcholů  $n = 4$  a  $x_{ij} = 1$ , a proto:

$$y_2 - y_3 + 4 \cdot 1 \leq 3,$$

$$y_3 - y_4 + 4 \cdot 1 \leq 3,$$

$$y_4 - y_2 + 4 \cdot 1 \leq 3,$$

po úpravě dostaneme nerovnice ve tvaru:

$$y_2 - y_3 \leq -1,$$

$$y_3 - y_4 \leq -1,$$

$$y_4 - y_2 \leq -1,$$

všechny tři rovnice se sečtou dohromady a vzniká nerovnice ve tvaru:

$$y_2 - y_3 + y_3 - y_4 + y_4 - y_2 \leq -3,$$

jelikož se všechna  $y$  mezi sebou odečtou, je výsledek nerovnice:

$$0 \leq -3,$$

nerovnice  $0 \leq -3$  neplatí, takže je splněna podmínka, že nedojde k uzavření vytvářeného cyklu.

Skupiny omezujících podmínek (4.1.5) a (4.1.6) se označují jako obligatorní podmínky, které charakterizují definiční obory  $x_{ij}$  a  $y_i$ .

## 4.2 Model č. 2 – obsluha letišť s časovými okny

Do druhého modelu, je kromě minimalizace délky celkové uletěné vzdálenosti v rámci okružního letu, zavedena časová proměnná, která umožní optimalizovat průběh letu v závislosti na “časových oknech“, která reprezentují časové intervaly nařazené pro obsluhu letišť.

### 4.2.1 Formulace problému – model č. 2

Je dána dopravní síť letových tras obsahující množinu letišť  $V$  ( $|V| = n$ ), ve které má probíhat zásilatelská činnost (svoz a rozvoz zásilek) leteckou dopravou. Analogicky, jako v modelu č. 1, představuje letiště  $v_i$  distribuční středisko, ze kterého bude probíhat obsluha. Je známa matice vzdáleností  $C$ , kde prvek  $c_{ij}$  reprezentuje délku minimální letové trasy letiště  $v_i$  pro  $i = 1, \dots, n$  na letiště  $v_j$  pro  $j = 1, \dots, n$ . Dále je pro každé z navštívených letišť  $v_j = 2, \dots, n$  znám časový interval tvořený dolní mezí  $d_j$  a horní mezí  $h_j$  reprezentující přípustnou dobu obsluhy zákazníka na letišti  $v_j$ . Hodnota  $Q$

reprezentuje čas strávený letadlem na letišti. Do tohoto časového intervalu jsou zahrnuty veškeré úkony probíhající mezi přistáním a startem letadla, např.: vykládka, nakládka zásilek, handling, tankování paliva, atd. Doba letu z letiště  $v_i$  na letiště  $v_j$  je vyjádřena časovou hodnotou  $T_{ij}$ , kde  $i = 1, \dots, n$  a  $j = 1, \dots, n$  a  $j \neq i$ , jejíž hodnoty se vypočítají ze vztahu:

$$T_{ij} = \frac{c(i,j)}{r} \cdot 60, \text{ pro } i = 1, \dots, n \text{ a } j = 1, \dots, n.$$

Dále model obsahuje prohibitivní konstantu  $M$ , což je hodnota sloužící ke správné funkci podmínek, ve kterých tato konstanta figuruje. Úkolem je opět naplánovat letovou trasu tak, aby v rámci jednoho okružního letu byla navštívena všechna letiště  $v_2, v_3, \dots, v_n$  a po jejich obsluze následoval návrat do střediska  $v_1$ .

Optimalizačním kritériem je celková vzdálenost, kterou letadlo v rámci obsluhy letišť absolvuje. Cílem optimalizace je minimalizace celkové vzdálenosti zohledňující optimalizaci průběhu letu v závislosti na časových oknech. Návaznost mezi jednotlivými letišti je opět zaručena bivalentní proměnnou  $x_{ij}$ . Je-li  $x_{ij} = 0$ , není spoj realizován, ale při  $x_{ij} = 1$  je spoj z letiště  $v_i$  na letiště  $v_j$  realizován. Dále je do modelu zavedena pomocná proměnná  $y_i$ , která bude použita v anticyklících podmínkách zaručujících, že nedojde k rozpadu vytvářeného okružního letu na více částí. Proměnná  $t_i$  představuje čas odletu z letiště  $v_i$ .

#### 4. 2. 2 Rekapitulace použitých zkratk – model č. 2

Konstanty:

- $c_{ij}$  délka minimální letové trasy z letiště  $v_i$  na letiště  $v_j$ ,  
pro  $i = 1, \dots, n$  a  $j = 1, \dots, n$ ,
- $n$  počet letišť,
- $Q$  doba strávená letadlem na letišti,
- $d_j$  dolní mez obsluhy letiště  $v_j$ , pro  $j = 1, \dots, n$ ,
- $h_j$  horní mez obsluhy letiště  $v_j$ , pro  $j = 1, \dots, n$ ,
- $T_{ij}$  doba letu z letiště  $v_i$  na letiště  $v_j$ , pro  $i = 1, \dots, n$  a  $j = 1, \dots, n$ ,
- $r$  průměrná rychlost letadla.



Proměnné:

- $x_{ij}$  proměnná modelující návaznost obsluhy jednotlivých letišť, *pro*  $i = 1, \dots, n$  a  $j = 1, \dots, n$ ,
- $y_i$  pomocná proměnná zavedená pro potřeby anticyklíci podmínky, *pro*  $i = 1, \dots, n$ ,
- $t_i$  čas odletu z letiště  $i = 1, \dots, n$ , *pro*  $i = 1, \dots, n$ .

#### 4. 2. 3 Matematický model č. 2

$$\min f(x, y, t) = \sum_{i=1}^n \sum_{j=1}^n c_{ij} \cdot x_{ij} \quad (4.2.1)$$

$$\sum_{i=1, i \neq j}^n x_{ij} = 1, \text{ pro } j = 1, \dots, n \quad (4.2.2)$$

$$\sum_{i=1, i \neq j}^n x_{ij} = \sum_{i=1, i \neq j}^n x_{ji}, \text{ pro } j = 1, \dots, n \quad (4.2.3)$$

$$y_i - y_j + n \cdot x_{ij} \leq n - 1, \text{ pro } i = 2, \dots, n \text{ a } j = 2, \dots, n \quad (4.2.4)$$

$$d_j \leq t_j - Q, \text{ pro } j = 1, \dots, n \quad (4.2.5)$$

$$h_j \geq t_j, \text{ pro } j = 1, \dots, n \quad (4.2.6)$$

$$Q + t_i + T_{ij} \leq t_j + M \cdot (1 - x_{ij}), \text{ pro } i = 1, \dots, n \text{ a } j = 2, \dots, n \text{ a } i \neq j \quad (4.2.7)$$

$$t_i \geq 0, \text{ pro } i = 1, \dots, n \quad (4.2.8)$$

$$x_{ij} \in \{0, 1\}, \text{ pro } i = 1, \dots, n \text{ a } j = 1, \dots, n, i \neq j \quad (4.2.9)$$

$$y_i \geq 0, \text{ pro } i = 2, \dots, n \quad (4.2.10)$$

#### 4. 2. 4 Komentář – model č. 2

Rovnice (4.2.1) je účelovou funkcí reprezentující celkovou délku absolvovaných letových tras a cílem optimalizace je minimalizace její hodnoty.

Skupina omezujících podmínek (4.2.2) zaručuje, že každý spoj bude obslužen právě jednou v rámci jednoho okružního letu. Skupina omezujících podmínek (4.2.3) zajistí kontinuitu organizace letu v rámci každého letiště. Dále skupina omezujících podmínek (4.2.4) představuje anticyklíci podmínku.

Skupiny omezujících podmínek (4.2.5) a (4.2.6) vymezují dolní a horní mez časového intervalu, ve kterém musí proběhnout odlet na letišti  $v_j$ . Podmínka (4.2.7) zaručuje přípustnou časovou posloupnost obsluhy letišť  $v_i$  a  $v_j$ . Podmínka (4.2.8) je obligatorní podmínka vztahující se k proměnné  $t_i$ . Skupiny obligatorních podmínek (4.2.9) a (4.2.10) charakterizují definiční obory proměnných  $x_{ij}$  a  $y_i$ .

### 4.3 Model č. 3 – obsluha letišť s časovými okny a proměnlivou dobou obsluhy

Vzhledem k tomu, že doba vykládky a nakládky zboží nebo velikost zásilky se může na jednotlivých letištích lišit, je potřeba pro větší přiblížení reálného provozu provést úpravy v modelu č. 2. Časová konstanta  $Q_j$  se bude lišit v závislosti na době strávené na letišti  $v_j$ . Proto je potřeba v modelu č. 2 upravit rovnice (4.2.7) a (4.2.9) následovně:

$$d_j \leq t_j - Q_j, \text{ pro } j = 2, \dots, n \quad (4.3.1)$$

$$Q_j + t_i + T_{ij} \leq t_j + M \cdot (1 - x_{ij}), \text{ pro } i = 1, \dots, n \text{ a } j = 2, \dots, n \text{ a } i \neq j \quad (4.3.2)$$

V rovnicích (4.3.1) a (4.3.2) došlo k úpravě konstanty  $Q$ , která v modelu č. 2 měla konstantní hodnotu stejnou pro obsluhu všech letišť. V modelu č. 3 se však konstanta  $Q$  bude lišit v závislosti na době obsluhy letiště  $v_j$ . Kromě této změny je model totožný s modelem č. 2.

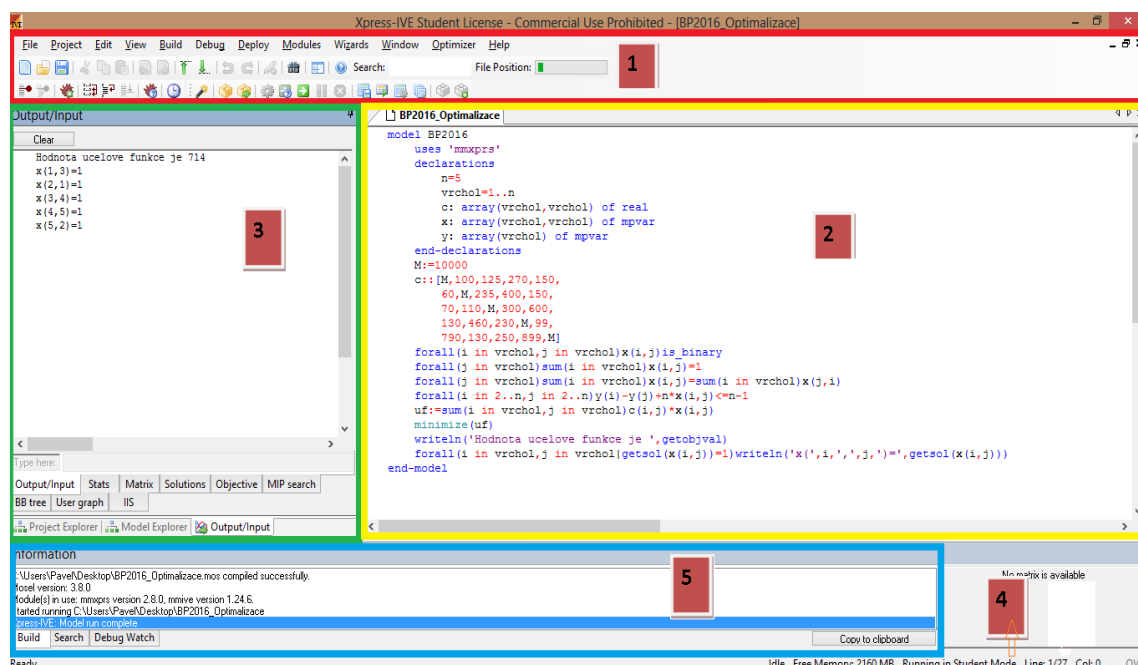
Cílem optimalizace je opět minimalizace celkové vzdálenosti zohledňující optimalizaci průběhu letu v závislosti na tzv. časových oknech.

## 5 Aplikační část – výpočetní experimenty s navrženými modely

V této kapitole předložené práce budou navržené matematické modely transformovány do programovacího jazyka Mosel a následně bude ověřena jejich funkčnost v optimalizačním software Xpress-IVE. Nejprve bude představen software Xpress-IVE, ve kterém budou následně provedeny modelové optimalizační výpočty.

### 5.1 Optimalizační software Xpress – IVE

Optimalizační software Xpress-IVE, je speciálně vyvinutý program pro řešení úloh matematického programování. Tento software využívá programovací jazyk Mosel a nabízí řešiteli jednoduché uživatelské rozhraní (obr. 10).



Obr. 10 – Pracovní prostředí programu Xpress-IVE [15]

Pracovní prostředí Xpress-IVE nabízí využití mnoha funkcí umístěných v horní liště (1). Pro práci s programem jsou důležité funkce Compile – sloužící k překladi textu programu během psaní modelu a současně i ke kontrole syntaktických chyb. Tyto chyby mohou vzniknout překlady v textu programu nebo špatně nadefinovaným modelem. Jestliže program odhalí chybu, vyznačí řádek s chybou žlutou barvou, jako na obr. 11, čímž upozorní uživatele na nutnost opravy.

```

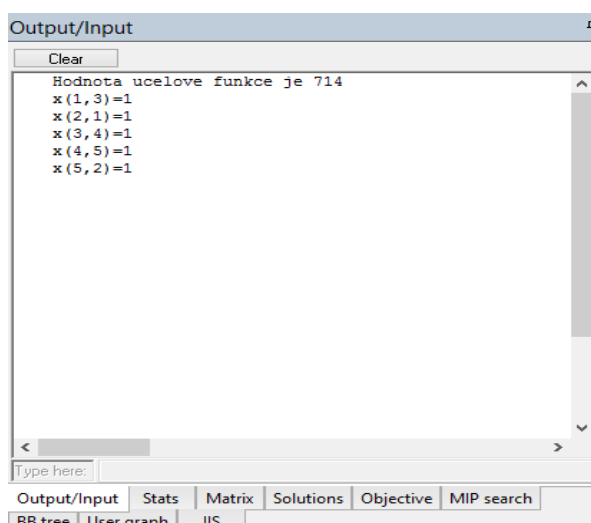
model BP2016
uses 'mmsxprs'
declarations
n=5
vrchol=1..n
c: array(vrchol, vrchol) of real
x: array(vrchol, vrchol) of mpvar
y: array(vrchol) of mpvar
end-declarations
M:=10000
c:=[M,100,125,270,150,
60,M,235,400,150,
70,110,M,300,600,
130,460,230,M,99,
790,130,250,899,M]
forall(i in vrchol, j in vrchol) x(i,j) is_binary
forall(j in vrchol) sum(i in vrchol) x(i,j)=1
forall(j in vrchol) sum(i in vrchol) x(i,j)=sum(i in vrchol) x(j,i)
forall(i in 2..n, j in 2..n) y(i)-y(j)+n*x(i,j) <= n-1k
uf:=sum(i in vrchol, j in vrchol) c(i,j)*x(i,j)
minimize(uf)
writeln('uf je ', getobjval)
forall(i in vrchol, j in vrchol | getsol(x(i,j))=1) writeln('x(', i, ', ', j, ')=' , getsol(x(i,j)))
end-model

```

Obr. 11 – Okno pro zadávání textu programu s upozorněním na chybu [15]

Na případné chyby je uživatel upozorněn také v dolní části obrazovky (5), kde jsou vypisována všechna upozornění na syntaktické chyby, které byly nalezeny. Jestliže po opravení nalezené chyby a opětovném stlačení tlačítka *Compile* nehlásí software žádnou chybu, lze přistoupit k optimalizačnímu výpočtu. Pomocí tlačítka *Run Model*, které je také umístěno na horní liště, lze zahájit optimalizační výpočet.

Prostřední část pracovní plochy lze rozdělit na dvě části. Pravé okno slouží pro zápis textu programu (2). V levé části (3) je okno, kde si uživatel může zvolit z několika záložek, jak je vidět na obr. 12. Z těchto záložek lze vyčíst téměř všechny potřebné údaje o průběhu optimalizačního výpočtu.



Obr. 12 – Okno s výslednými daty v Xpress-IVE [15]

V záložce *Output/Input* se uživateli zobrazí informace, které požaduje vypsát. Záložka *Stats* slouží k zobrazení hodnoty účelové funkce a dalších stavových hlášení souvisejících s průběhem optimalizačního výpočtu. V záložce *Solutions* jsou zobrazeny výsledné hodnoty proměnných vyskytujících se v modelu. Grafy, popisující časový průběh hledání optimálního řešení, jsou v záložce *MIP search*. Pod záložkou *BB tree* se nachází strom řešení, jestliže jsou v modelu využívány bivalentní nebo celočíselné nezáporné proměnné. V levé části se dále nachází okno, kde se vypisují údaje o všech entitách vyskytujících se v modelu, po ukončení i hodnoty jednotlivých proměnných. Další informace (volná výpočetní paměť programu, poloha kurzoru v příkazovém řádku) jsou zobrazeny na spodní liště (4).

## 5.2 Optimalizační výpočty v software Xpress - IVE

Než budou provedeny samotné výpočetní experimenty, budou navržené modely transformovány do programovacího jazyku MOSEL. Transformovaný model lze rozdělit do několika částí – název modelu, definice použitých modulů, deklarační část, proměnné a konstanty, soustava omezujících podmínek, účelová funkce, sdělení a ukončení modelu.

Jako první musí být definován název modelu a používané moduly:

**model BP20161** – *název programu*

**uses 'mmxprs'** – *použité moduly*

Dále musí být zapsána deklarační část, která je zahájena klíčovým slovem – *declarations* a ukončena klíčovým souslovím – *end-declarations*. V deklarační části jsou zadefinovány všechny množiny, konstanty (*real*) a proměnné (*mpvar*), které se v modelu vyskytují.

**declarations**

**n=5** – *počet prvků množiny*

**vrchol=1..n** – *deklarace množiny vrcholů V*

**c: array(vrchol, vrchol) of real** – *deklarace konstant použitých v modelu*

**x: array(vrchol, vrchol) of mpvar** – *deklarace rozhodovacích proměnných použitých v modelu*

**y: array(vrchol) of mpvar** – *deklarace pomocných proměnných použitých v modelu*

### **end-declarations**

Po ukončení deklarační části jsou definovány vstupní hodnoty všech konstant uvedených v této úloze.

**M:=10000**

**c::[M,100,125,270,150,**

**60,M,235,400,150,**

**70,110,M,300,600,**

**130,460,230,M,99,**

**790,130,250,899,M]**

Následuje sestavení omezujících podmínek, účelové funkce a požadavek na hledaný extrém účelové funkce.

**forall(i in vrchol,j in vrchol)x(i,j)is\_binary**

**forall(j in vrchol)sum(i in vrchol)x(i,j)=1**

**forall(j in vrchol)sum(i in vrchol)x(i,j)=sum(i in vrchol)x(j,i)**

**forall(i in 2..n,j in 2..n)y(i)-y(j)+n\*x(i,j)<=n-1**

**uf:=sum(i in vrchol,j in vrchol)c(i,j)\*x(i,j)**

**minimize(uf)**

V závěru může uživatel požádat o výpis výsledků (např. hodnotu účelové funkce) v záložce *Output/Input*.

**writeln(,Hodnota ucelove funkce je „getobjval)**

Pro ukončení modelu je do příkazového okna zapsána fráze *end-model* pro ukončení.

**end-model**

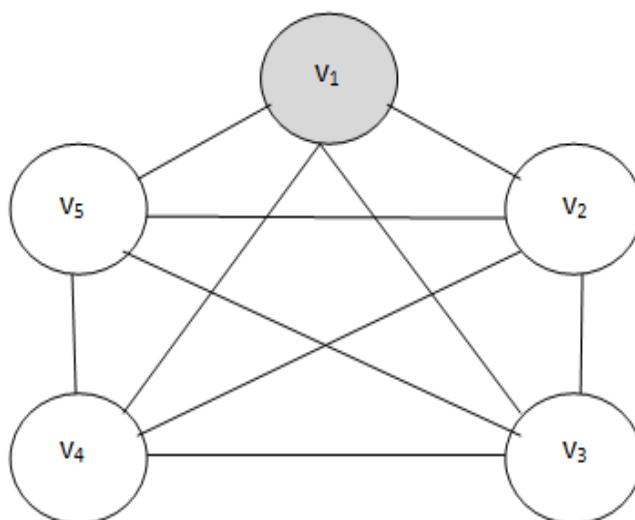
Jestliže si chce uživatel do textu programu vložit poznámku, musí použít symbol *“!‘‘*. Při psaní textu za tento symbol, nedojde k ovlivnění samotného textu modelu.

Nyní bude následovat ověření funkčnosti navržených modelů pro řešení optimalizace letové trasy v rámci obsluhy letišť z jednoho střediska. V předložené bakalářské práci se jedná pouze o modelové příklady.

### 5.3 Výpočetní experiment – model č. 1

Je dána síť letových tras, ve které je úkolem naplánovat zasílatelskou činnost. Dopravní síť obsahuje množinu letišť  $V$  ( $|V| = n$ ), kde  $n = 5$ . Letiště  $v_1$  představuje distribuční středisko, ze kterého bude obsluha probíhat. Dále je známa matice vzdáleností  $C$ , přičemž prvek  $c_{ij}$  reprezentuje délku minimální letové trasy z letiště  $v_i$  na letiště  $v_j$  (viz tab. 1).

Úkolem je naplánovat letovou trasu tak, aby byla v rámci jednoho okružního letu obsloužena všechna letiště, a aby celková absolvovaná vzdálenost byla minimální. Zároveň musí být splněna podmínka, že letiště  $v_1$  představuje počáteční a koncový bod letové trasy. Na obr. 13 jsou zobrazeny všechny možnosti obsluhy z letiště  $v_1$ .



Obr. 13 – Možnosti obsluhy z letiště  $v_1$  – model č. 1

Vstupní údaje:

$c_{ij}$	1	2	3	4	5
1	M	100	125	270	150
2	60	M	235	400	150
3	70	110	M	300	600
4	130	460	230	M	99
5	790	130	250	899	M

Tab. 1 – Matice vzdáleností  $c_{ij}$  pro model č. 1

### 5. 3. 1 Text programu – model č. 1

model BP2016

```
uses 'mmxprs'

declarations

    n=5

    vrchol=1..n

    c: array(vrchol,vrchol) of real

    x: array(vrchol,vrchol) of mpvar

    y: array(vrchol) of mpvar

end-declarations

M:=10000

c::[M,100,125,270,150,
    60,M,235,400,150,
    70,110,M,300,600,
    130,460,230,M,99,
    790,130,250,899,M]

forall(i in vrchol,j in vrchol)x(i,j)is_binary
forall(j in vrchol)sum(i in vrchol)x(i,j)=1
forall(j in vrchol)sum(i in vrchol)x(i,j)=sum(i in vrchol)x(j,i)
forall(i in 2..n,j in 2..n)y(i)-y(j)+n*x(i,j)<=n-1

uf:=sum(i in vrchol,j in vrchol)c(i,j)*x(i,j)

minimize(uf)

writeln('Hodnota ucelove funkce je ',getobjval)

forall(i in vrchol,j in vrchol|getsol(x(i,j))=1)writeln('x(','i','j,')=' ,getsol(x(i,j)))

end-model
```

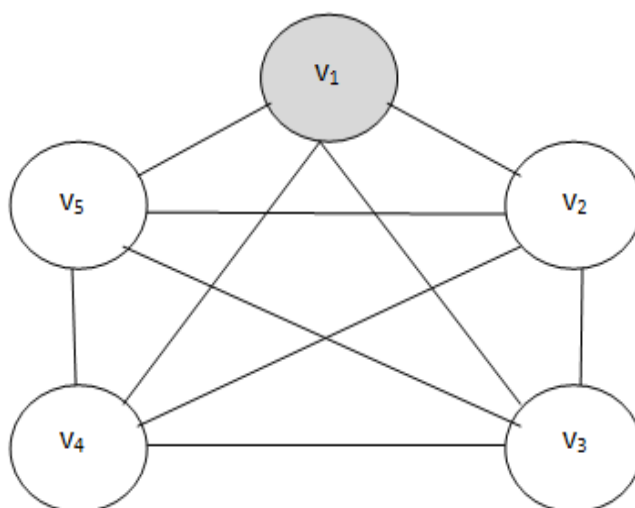
### 5. 4 Výpočetní experiment – model č. 2

Je dána síť letových tras, ve které je úkolem naplánovat zasílatelskou činnost. Dopravní síť obsahuje množinu letišť  $V$  ( $|V| = n$ ), kde  $n = 5$ . Letiště  $v_1$  představuje



distribuční středisko, ze kterého bude obsluha probíhat. Dále je známa matice vzdáleností  $C$ , přičemž prvek  $c_{ij}$  reprezentuje délku minimální letové trasy z letiště  $v_i$  na letiště  $v_j$  (viz tab. 2). Dále je známý časový interval tvořený dolní mezí  $d_j$  a horní mezí  $h_j$  (viz tab. 3), ze kterého nabývá hodnot proměnná  $t_i$ . Do úlohy je zavedená časová konstanta  $Q = 60$  minut, která představuje čas strávený letadlem na letišti pro potřeby vykládky, nakládky zboží a na handling. Pro zvolenou úlohu je do modelu zavedená rychlost letadla  $r = 300$  km/h.

Úkolem je opět naplánovat letovou trasu tak, aby byla v rámci jednoho okružního letu obsloužena všechna letiště. Cílem úlohy je minimalizace celkové vzdálenosti zohledňující závislost na “časových oknech”. Na obr. 14 jsou zobrazeny všechny možnosti obsluhy.



Obr. 14 – Možnosti obsluhy z letiště  $v_1$  – model č. 2

Vstupní údaje:

$c_{ij}$	1	2	3	4	5
1	M	100	125	270	150
2	60	M	235	400	150
3	70	110	M	300	600
4	130	460	230	M	99
5	790	130	250	899	M

Tab. 2 – Matice vzdálenosti  $c_{ij}$  pro model č. 2

Letiště	1	2	3	4	5
$d_j$	0	200	160	900	700
$h_j$	M	560	560	1050	999

Tab. 3 – Časové intervaly pro odlet z letiště  $v_j$  – model č. 2

### 5. 4. 1 Text programu – model č. 2

model BP2016\_2

uses 'mmxprs'

declarations

n=5

vrchol=1..n

c: array(vrchol, vrchol) of real

d: array(vrchol) of real

h: array(vrchol) of real

x: array(vrchol, vrchol) of mpvar

y: array(vrchol) of mpvar

t: array(vrchol) of mpvar

T: array(vrchol, vrchol) of real

end-declarations

M:=10000

r:=300

Q:=60

c::[M,100,125,270,150,  
60,M,235,400,150,  
70,110,M,300,600,  
130,460,230,M,99,  
790,130,250,899,M]

forall(i in vrchol,j in vrchol)T(i,j):=c(i,j)/r\*60

d::[0,200,160,900,700]

h::[M,560,560,1050,999]

forall(j in vrchol)d(j)<=t(j)-Q

forall(j in vrchol)h(j)>=t(j)

forall(i in vrchol,j in 2..5)t(i)+T(i,j)+Q<=t(j)+M\*(1-x(i,j))

```

forall(i in vrchol,j in vrchol)x(i,j)is_binary
forall(j in vrchol)sum(i in vrchol)x(i,j)=1
forall(j in vrchol)sum(i in vrchol)x(i,j)=sum(i in vrchol)x(j,i)
forall(i in 2..n,j in 2..n)y(i)-y(j)+n*x(i,j)<=n-1
uf:=sum(i in vrchol,j in vrchol)c(i,j)*x(i,j)
minimize(uf)

writeln('Hodnota ucelove funkce je ',getobjval)

forall(i in vrchol,j in vrchol|getsol(x(i,j))=1)writeln('x(' ,i ,',' ,j ,')=' ,getsol(x(i,j)))

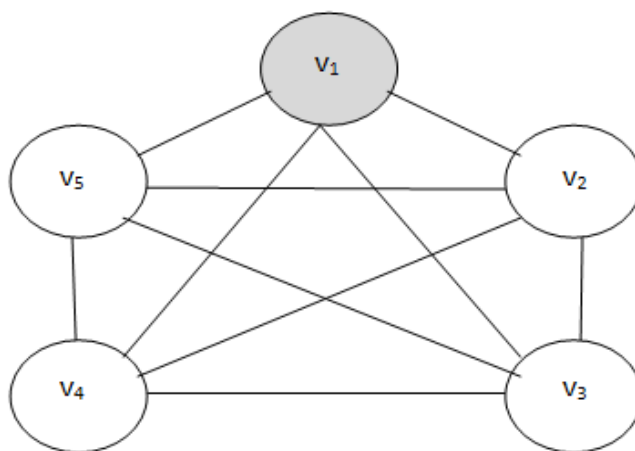
end-model

```

## 5.5 Výpočetní experiment – model č. 3

Je dána síť letových tras, ve které je úkolem naplánovat zásílatelskou činnost. Dopravní síť obsahuje množinu letišť  $V$  ( $|V| = n$ ), kde  $n = 5$ . Matice vzdáleností  $C$  s prvky  $c_{ij}$ , je uvedena v tab. 4. Oproti modelu č. 2 je změněna časová konstanta  $Q$ , která představuje čas strávený letadlem na letišti pro potřeby vykládky, nakládky zboží a na handling. Hodnoty konstanty  $Q_j$  jsou uvedeny v tab. 5. Časový interval tvořený dolní mezí  $d_j$  a horní mezí  $h_j$  je uveden v tab. 6. Pro zvolenou úlohu je do modelu zavedená rychlost letadla  $r = 200$  km/h.

Úkolem je opět naplánovat letovou trasu tak, aby byla v rámci jednoho okružního letu obsloužena všechna letiště. Cílem úlohy je minimalizace celkové vzdálenosti zohledňující závislost na “časových oknech”. Na obr. 15 jsou zobrazeny všechny možnosti obsluhy.



Obr. 15 – Možnosti obsluhy z letiště  $v_1$  – model č. 3

Vstupní údaje:

$c_{ij}$	1	2	3	4	5
1	M	100	125	270	150
2	60	M	235	400	150
3	70	110	M	300	600
4	130	460	230	M	99
5	790	130	250	899	M

Tab. 4 – Matice vzdálenosti  $c_{ij}$  pro model č. 3

Letiště	1	2	3	4	5
$Q_j$	35	45	55	65	75

Tab. 5 – Matice hodnot konstanty  $Q_j$  pro model č. 3

Letiště	1	2	3	4	5
$d_j$	-	150	280	400	700
$h_j$	-	900	345	1050	900

Tab. 6 – Časové intervaly pro odlet z letiště  $v_j$  – model č. 3

### 5. 5. 1 Text programu – model č. 3

model BP2016\_3

uses 'mmxprs'

declarations

n=5

vrchol=1..n

Q: array(vrchol) of real

c: array(vrchol,vrchol) of real

d: array(2..n) of real

h: array(2..n) of real

x: array(vrchol,vrchol) of mpvar

y: array(vrchol) of mpvar

t: array(vrchol) of mpvar

T: array(vrchol,vrchol) of real

end-declarations

M:=10000

```

r:=200
Q::[35,45,55,65,75]
c::[M,100,125,270,150,
    60,M,235,400,150,
    70,110,M,300,600,
    130,460,230,M,99,
    790,130,250,899,M]
d::[150,280,400,700]
h::[900,345,1050,900]
forall(i in vrchol,j in vrchol)T(i,j):=c(i,j)*60/r
forall(j in 2..n)d(j)<=t(j)-Q(j)
forall(j in 2..n)h(j)>=t(j)
forall(i in vrchol,j in 2..5)t(i)+T(i,j)+Q(j)<=t(j)+M*(1-x(i,j))
forall(i in vrchol,j in vrchol)x(i,j)is_binary
forall(j in vrchol)sum(i in vrchol)x(i,j)=1
forall(j in vrchol)sum(i in vrchol)x(i,j)=sum(i in vrchol)x(j,i)
forall(i in 2..n,j in 2..n)y(i)-y(j)+n*x(i,j)<=n-1
uf:=sum(i in vrchol,j in vrchol)c(i,j)*x(i,j)
minimize(uf)
writeln('Hodnota ucelove funkce je ',getobjval)
forall(i in vrchol,j in vrchol|getsol(x(i,j))=1)writeln('x(' ,i ,',' ,j ,')=' ,getsol(x(i,j)))
end-model

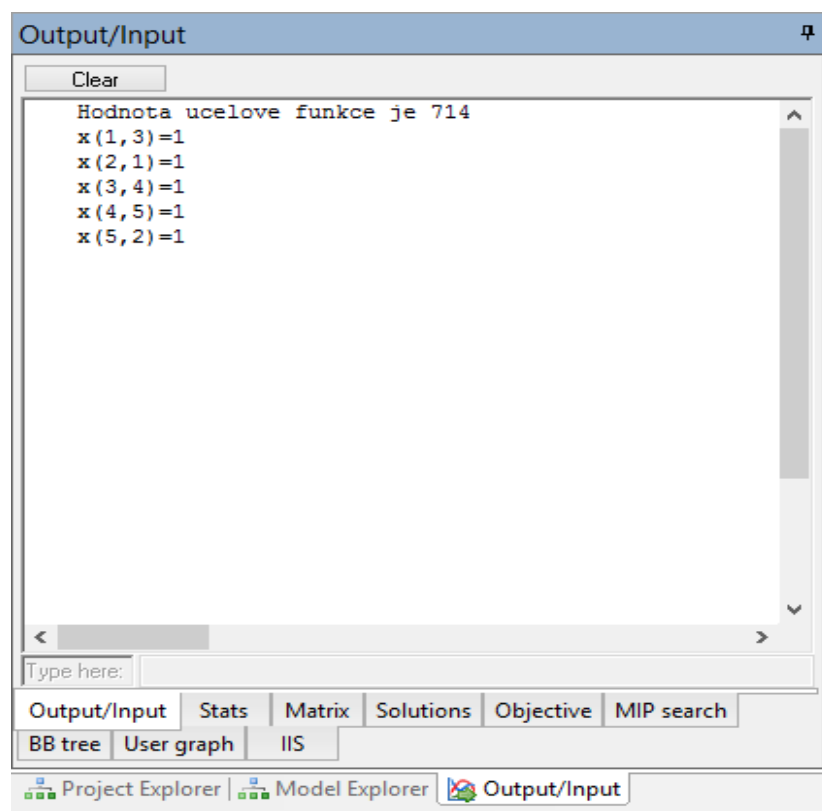
```

## 6 Zhodnocení dosažených výsledků

V následujících kapitolách 6. 1 a 6. 2 a 6. 3 budou vyhodnoceny výsledky modelu č. 1, modelu č. 2 a modelu č. 3.

### 6. 1 Interpretace výsledků – model č. 1

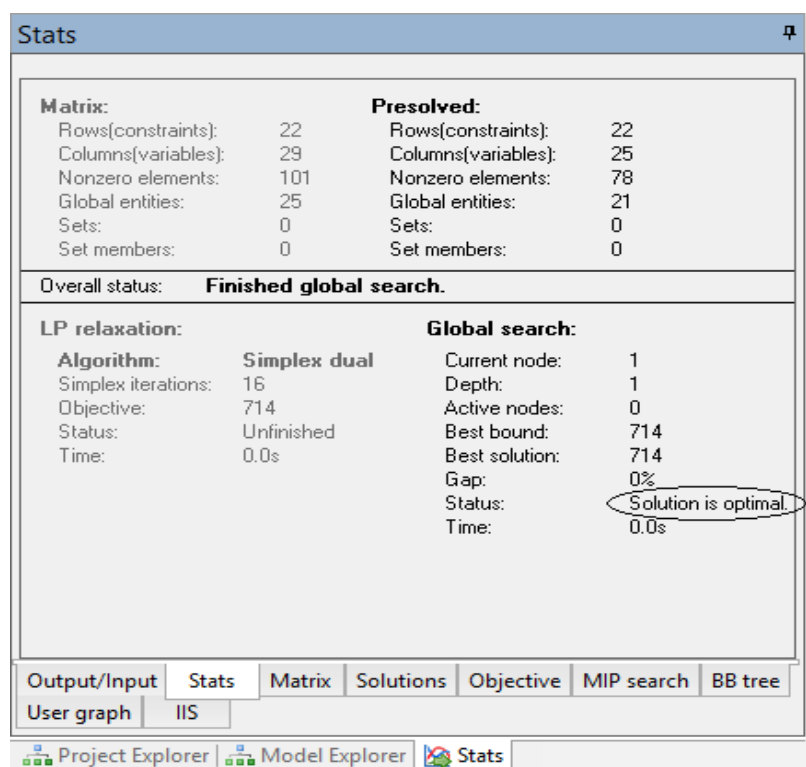
Výstup z modelu č. 1 řešeného v software Xpress-IVE je zobrazen na obr. 16.



Obr. 16 – Hodnota účelové funkce a návaznosti letu v modelu č. 1 [15]

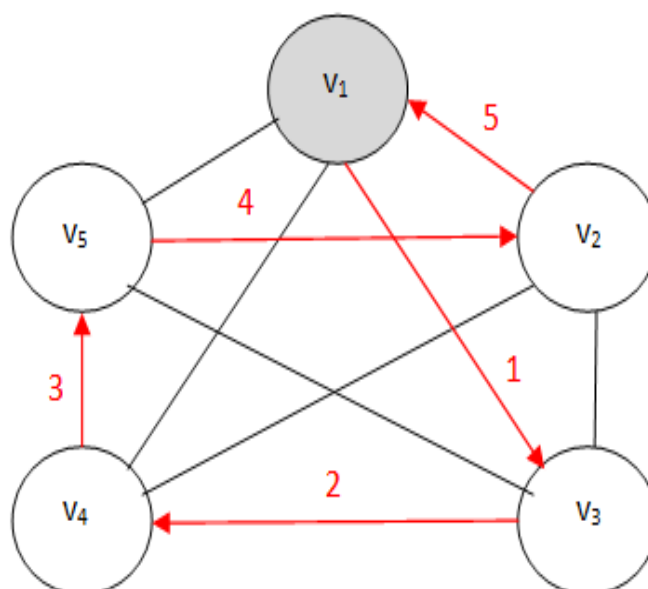
V okně Output/Input jsou zobrazeny veškeré informace, které byly požadovány vypsat v textu programu. Algoritmus vypočítal, že hodnota účelové funkce je 714 km, což představuje celkovou absolvovanou vzdálenost. Dále jsou zde vypsány proměnné  $x_{ij}$ , které nabývají hodnoty 1. Lze tedy vyčíst, že letová trasa je: 1 – 3 – 4 – 5 – 2 – 1.

Na obr. 17 je uveden výpis statistických údajů v software Xpress-IVE. U vyznačeného údaje *Status* se nachází informace, že vyhledané řešení je optimální. Údaj *Time*, nacházející se pod údajem *Status*, značí celkovou dobu výpočtu – 0 s, což znamená, že program tuto úlohu vyřešil opravdu rychle.



Obr. 17 – Statistické údaje o výpočtu modelu č. 1 [15]

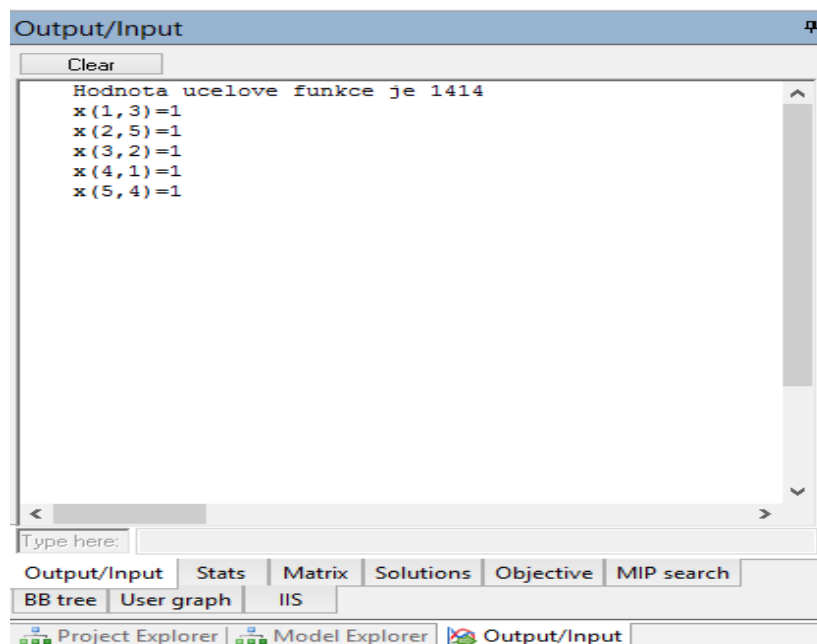
Na obr. 18 je zakreslena letová trasa modelu č. 1: 1 – 3 – 4 – 5 – 2 – 1. Vypočtená letová trasa je optimální. Její celková délka je 714 km.



Obr. 18 – Letová trasa modelu č. 1

## 6.2 Interpretace výsledků – model č. 2

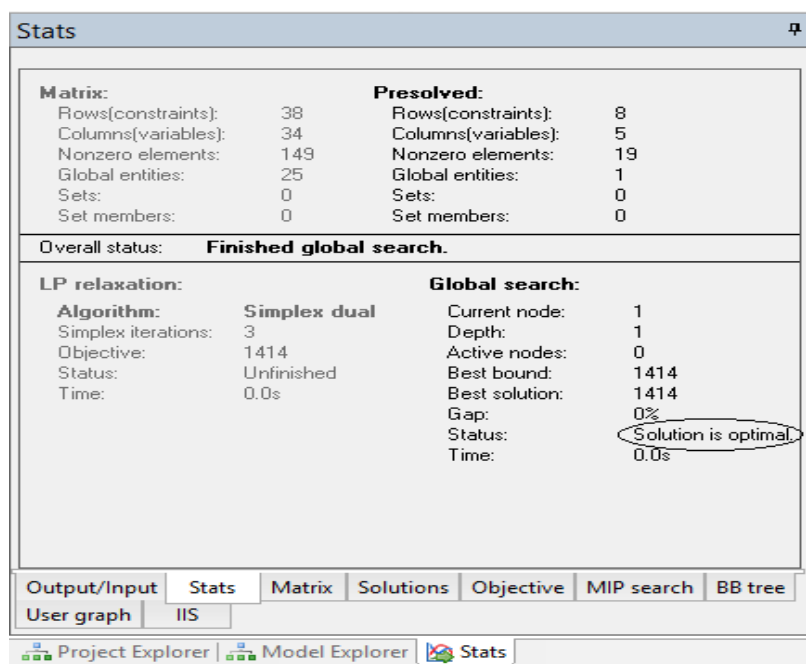
Výstup z modelu č. 2 řešeného v programu Xpress-IVE je zobrazen na obr. 19.



Obr. 19 - Hodnota účelové funkce a návaznosti letu v modelu č. 2 [15]

Hodnota účelové funkce, tedy celková absolvovaná vzdálenost v modelu č. 2 je 1 414 km. Opět jsou zde vypsány hodnoty proměnných  $x_{ij}$  modelujících uskutečnění přesunů mezi letišti. Naplánovaná letová trasa na základě modelu č. 2 je: 1 – 3 – 2 – 5 – 4 – 1.

Na obr. 20 jsou zobrazeny statistické údaje z výpočtu modelu č. 2.



Obr. 20 - Statistické údaje o výpočtu modelu č. 2 [15]

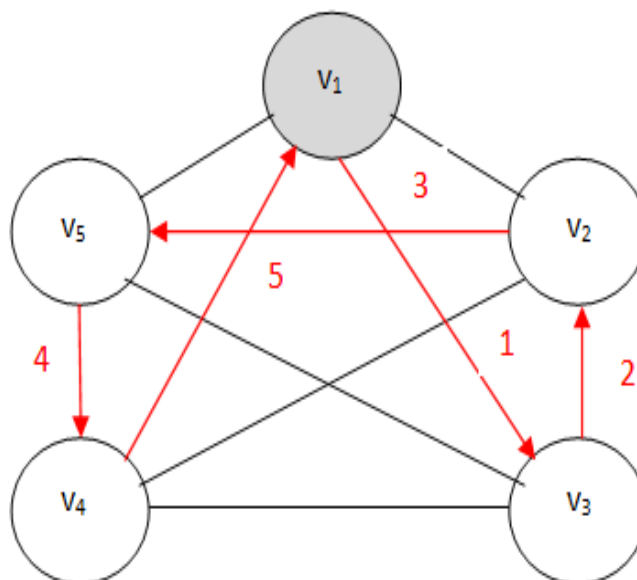


Vyhledané řešení je znovu optimální, což je zapsáno u kolonky *Status: Solution is optimal*. Čas výpočtu je opět velmi rychlý, jelikož činí 0 sekund. V tab. 7 jsou v kolonce  $t_i$  uvedeny časy odletů z jednotlivých letišť.

letišťe	$d_j$	$h_j$	$t_i$
1	0	M	60
2	200	560	302
3	160	560	220
4	900	1050	1050
5	700	999	760

Tab. 7 – Časy odletů z jednotlivých letišť pro model č. 2

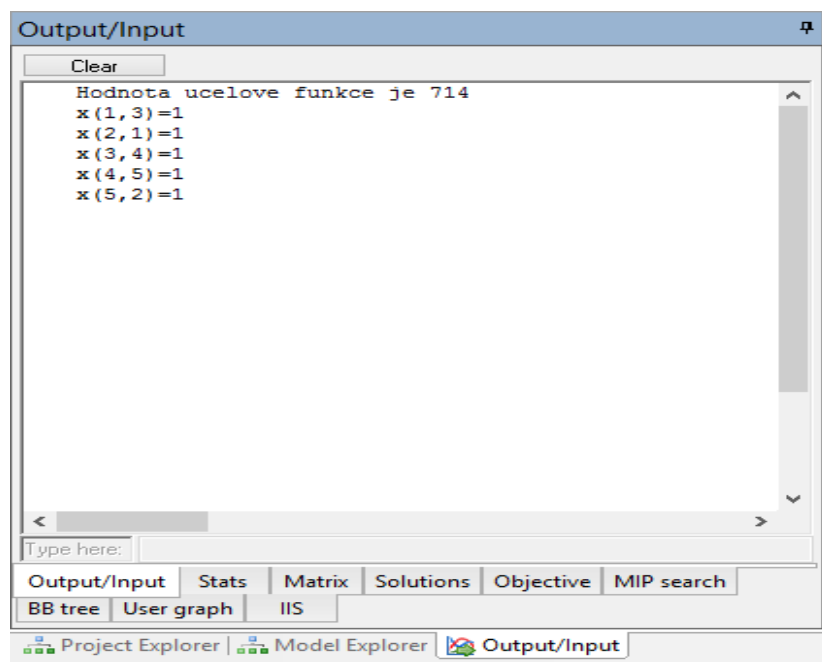
Naplánovaná letová trasa 1 – 3 – 2 – 5 – 4 – 1 je zobrazena na obr. 21. Její celková délka je 1 414 km a splňuje podmínku, že počáteční i koncový bod trasy se nachází ve středisku  $v_I$ .



Obr. 21 – Letová trasa modelu č. 2

### 6.3 Interpretace výsledků – model č. 3

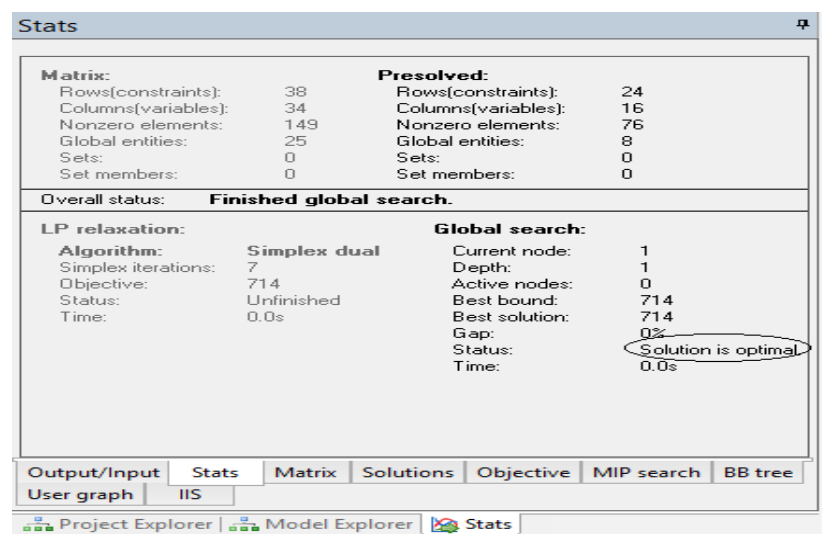
Výstup z modelu č. 3 řešeného v software Xpress-IVE je zobrazen na obr. 22.



Obr. 22 - Hodnota účelové funkce a návaznosti letu v modelu č. 3 [15]

Účelová funkce, představující celkovou absolvovanou vzdálenost, má pro model č. 3 hodnotu 714 km. Jsou zde stejně jako v předchozích modelech vypsány hodnoty proměnných  $x_{ij}$ , u kterých je spoj realizován. Naplánovaná trasa modelu č. 3 je: 1 – 3 – 4 – 5 – 2 – 1.

Na obr. 23 jsou uvedeny statistické údaje o výpočtu z modelu č. 3, který byl proveden v software Xpress-IVE.



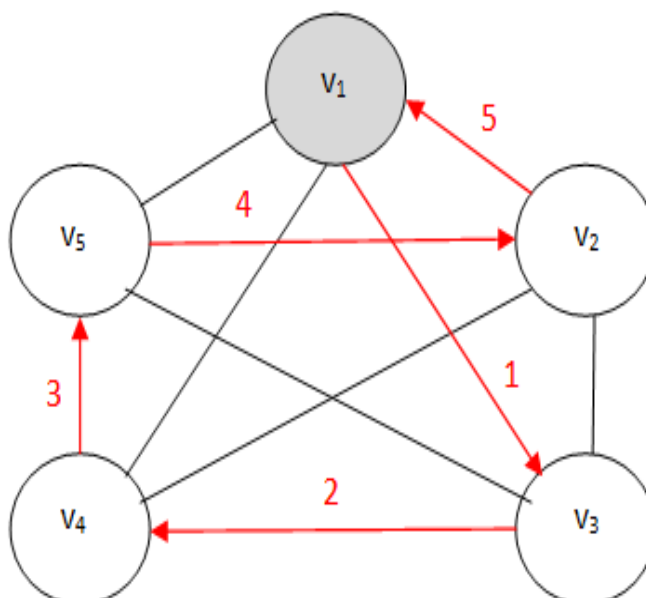
Obr. 23 - Statistické údaje o výpočtu modelu č. 3 [15]

Vyhledané řešení je optimální, což je vyznačeno elipsou u kolonky *Status*. Čas výpočtu je opět 0 sekund. V tab. 8 jsou v kolonce  $t_i$  uvedeny časy odletů z jednotlivých letišť, přičemž z tabulky lze vyčíst, že časy odletů jsou v rozmezí dolní meze  $d_j$  a horní meze  $h_j$ .

letišťe	$d_j$	$h_j$	$t_i$
1	–	–	0
2	150	900	859
3	280	345	335
4	400	1050	490
5	700	900	775

Tab. 8 – Časy odletů z jednotlivých letišť pro model č. 3

Naplánovaná letová trasa 1 – 3 – 4 – 5 – 2 – 1 je zobrazena na obr. 24. Její celková délka je 714 km a splňuje podmínku, že počáteční i koncový bod trasy se nachází ve středisku - letišti  $v_1$ .



Obr. 24 – Letová trasa modelu č. 3

## Závěr

Cílem bakalářské práce bylo interpretovat a navrhnout matematické modely na bázi lineárního programování, sloužící pro plánování obsluhy letišť z jednoho střediska. Úvodní kapitoly jsou zaměřeny na přípravu a plánování letů, na teoretické znalosti z oblasti optimalizačních metod a matematického programování, potřebné pro pochopení dané problematiky.

Dále byly prezentovány tři matematické modely, z nichž první uvažuje obsluhu letišť z jednoho střediska bez časových oken, druhý model uvažuje obsluhu letišť z jednoho střediska s časovými okny a třetí model uvažuje obsluhu letišť z jednoho střediska s časovými okny a proměnlivou dobou obsluhy. Ve všech případech je zvoleno středisko – letiště  $v_I$ , které je počátečním a zároveň koncovým bodem letové trasy. Pro všechny modely je zvoleno pět letišť, ve kterých obsluha probíhá. Modely jsou koncipovány tak, aby byla obsloužena všechna letiště. Optimalizačním kritériem ve všech třech modelech je minimalizace celkové absolvované vzdálenosti.

Po sestavení matematických modelů následovala transformace těchto modelů do programovacího jazyka MOSEL. V optimalizačním software Xpress-IVE byla ověřena jejich funkčnost. Pro zpracování byl využit počítač HP ProBook 4340s s procesorem Intel CORE i3. Následovalo zhodnocení dosažených výsledků, kde byly interpretovány výsledky modelu č. 1, kde byla celková absolvovaná vzdálenost 714 km, v modelu č. 2 byla vzdálenost 1 414 km a v modelu č. 3 byla délka trasy 714 km. Ve všech třech úlohách bylo nalezeno optimální řešení a podařilo se tedy najít optimální letovou trasu z hlediska celkové absolvované vzdálenosti.

Lze tedy říci, že matematické programování je možné aplikovat v letecké dopravě k potřebám optimalizace. V této práci se jedná o optimalizaci pouze pro modelové příklady, a proto by bylo pro budoucí řešitele vhodné pracovat s reálnými údaji, kde by výsledky optimalizačních výpočtů mohly být ověřeny v reálném provozu.

## Použitá literatura

- [1] JANÁČEK, Jaroslav. *Matematické programování*. 2. opr. vyd. Žilina: Žilinská univerzita, 2003. ISBN 80-8070-054-0.
- [2] JABLONSKÝ, Josef. *Operační výzkum: kvantitativní modely pro ekonomické rozhodování*. 1. vyd. Praha: Professional Publishing, 2002. ISBN 80-86419-23-1.
- [3] BÍNA, Ladislav. *Provozování letecké dopravy a logistika*. Vyd. 1. Brno: Akademické nakladatelství CERM, 2014. ISBN 978-80-7402-855-7.
- [4] DANĚK, Jan, TEICHMANN, Dušan. *Optimalizace dopravních procesů*. 1. vyd. Ostrava: VŠB - Technická univerzita Ostrava, 2005. ISBN 80-248-0996-6.
- [5] DORDA, Michal. *1. část: Základy teorie grafů* [online]. [cit. 2016-03-25]. Dostupné z: [http://homel.vsb.cz/~dor028/Teorie\\_grafu.pdf](http://homel.vsb.cz/~dor028/Teorie_grafu.pdf)
- [6] TEICHMANN, Dušan. Studijní opory k modulu M14 - Optimalizace technologických procesů. [online]. ©2012 [25-3-2016]. <http://www.vvvd.cz/m14-optimalizace-technologickych-procesu-29.html>
- [7] DORDA, Michal. *Littlův algoritmus* [online]. 2012. [cit. 2016-03-25]. Dostupné z: [https://issuu.com/michdor/docs/m14\\_animace\\_5?e=7481937/1637232](https://issuu.com/michdor/docs/m14_animace_5?e=7481937/1637232)
- [8] DORDA, Michal. *2. část: Základy matematického programování, dopravní úloha*. [online]. [cit. 2016-03-25]. Dostupné z: [http://homel.vsb.cz/~dor028/Dopravni\\_uloha.pdf](http://homel.vsb.cz/~dor028/Dopravni_uloha.pdf)
- [9] LAMPA, Martin, ZAPLETAL, František, LENORT, Radim, WICHER, Pavel a KRAUSOVÁ, Emilie. *Exaktní metody rozhodování I* [CD-ROM]. 1. vydání. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2015 [cit. 2016-03-17]. ISBN 978-80-248-3729-1.
- [10] PRUŠA, Jiří, a kol. *Svět letecké dopravy*. 1. vyd. Praha: Galileo CEE Service ČR, 2007. 315 s. ISBN 978-80-239-9206-9. FRYNTA, Jiří, Jiří LOUBAL a Jan SCHOŘ.
- [11] *Plánování letu a monitorování letu (033)*. Vyd. 1. Brno: Akademické nakladatelství CERM, 2007. ISBN 978-80-7204-501-3.
- [12] Studijní materiály z předmětů Plánování a provedení letu 1, 2, 3, 4
- [13] Studijní materiály z předmětu Provoz a ekonomika letecké dopravy 1

- [14] EISLER, Jan. *Úvod do ekonomiky dopravy*. Vyd. 1. Praha: Codex Bohemia, 1998. ISBN 80-85963-54-X.
- [15] FAIR ISAAC CORPORATIN. [online]. ©2016 [25-3-2016]. <http://www.fico.com/en/products/fico-xpress-optimization-suite/>
- [16] BRAMEL, Julien, SIMCHI-LEVI, David. *The logic of logistics: theory, algorithms, and applications for logistics management*. New York: Springer, c1997. ISBN 0-387-94921-6.
- [17] URANOS Flight Planning System. *IFR Flight Planning System*. [online]. 2013 [cit. 2016-03-20]. Dostupné z: <http://www.nav.cz/#NaN>
- [18] Hub and spoke. . [online]. [cit. 2016-03-20]. Dostupné z: [https://www.google.cz/search?hl=cs&site=imghp&tbm=isch&source=hp&biw=1366&bih=643&q=hub+and+spoke&oq=Hub+a&gs\\_l=img.1.0.0j0i30l9.124.1748.0.3182.10.7.2.0.0.0.139.746.4j3.7.0....0...1ac.1.64.img..1.9.762.0Rrn8Xst-S8#imgrc=4NupcNpR9rFkSM%3A-hub+and+spoke](https://www.google.cz/search?hl=cs&site=imghp&tbm=isch&source=hp&biw=1366&bih=643&q=hub+and+spoke&oq=Hub+a&gs_l=img.1.0.0j0i30l9.124.1748.0.3182.10.7.2.0.0.0.139.746.4j3.7.0....0...1ac.1.64.img..1.9.762.0Rrn8Xst-S8#imgrc=4NupcNpR9rFkSM%3A-hub+and+spoke)

## Seznam obrázků

- Obr. 1 – Základní schéma leteckého přepravního systému [3], s. 12
- Obr. 2 – Úsek letové trati L984\*, s. 13
- Obr. 3 – Princip technologie point to point [13], s. 13
- Obr. 4 – Princip technologie Hub and spoke [18], s. 13
- Obr. 5 – Schéma systému plánování a přípravy letu [3], s. 15
- Obr. 6 – Ukázka pracovního prostředí NAVairways [17], s. 16
- Obr. 7 – Graf s ohodnocením hran, s. 19
- Obr. 8 – Hamiltonova kružnice [7], s. 21
- Obr. 9 – Obsluha letišť ze střediska  $v_1$ , s. 28
- Obr. 10 – Pracovní prostředí programu Xpress-IVE [15], s. 35
- Obr. 11 – Okno pro zadávání textu programu s upozorněním na chybu [15], s. 36
- Obr. 12 – Okno s výslednými daty v Xpress-IVE [15], s. 36
- Obr. 13 – Možnosti obsluhy z letiště  $v_1$  – model č. 1, s. 39
- Obr. 14 – Možnosti obsluhy z letiště  $v_1$  – model č. 2, s. 41
- Obr. 15 – Možnosti obsluhy z letiště  $v_1$  – model č. 3, s. 43
- Obr. 16 – Hodnota účelové funkce a návaznosti letu v modelu č. 1 [15], s. 46
- Obr. 17 – Statistické údaje o výpočtu modelu č. 1 [15], s. 47
- Obr. 18 – Letová trasa modelu č. 1, s. 47
- Obr. 19 - Hodnota účelové funkce a návaznosti letu v modelu č. 2 [15], s. 48
- Obr. 20 - Statistické údaje o výpočtu modelu č. 2 [15], s. 48
- Obr. 21 – Letová trasa modelu č. 2, s. 49
- Obr. 22 - Hodnota účelové funkce a návaznosti letu v modelu č. 3 [15], s. 50

Obr. 23 - Statistické údaje o výpočtu modelu č. 3 [15], s. 50

Obr. 24 – Letová trasa modelu č. 3, s. 51



## Seznam tabulek

Tab. 1 – Matice vzdálenosti  $c_{ij}$  pro model č. 1, s. 39

Tab. 2 – Matice vzdálenosti  $c_{ij}$  pro model č. 2, s. 41

Tab. 3 – Časové intervaly pro odlet z letiště  $v_j$  – model č. 2, s. 41

Tab. 4 – Matice vzdálenosti  $c_{ij}$  pro model č. 3, s. 44

Tab. 5 – Matice hodnot konstanty  $Q_j$  pro model č. 3, s. 44

Tab. 6 – Časové intervaly pro odlet z letiště  $v_j$  – model č. 3, s. 44

Tab. 7 – Časy odletů z jednotlivých letišť pro model č. 2, s. 49

Tab. 8 – Časy odletů z jednotlivých letišť pro model č. 3, s. 51